ABRIL 1986 — Nº 13 A 2 REP. ARGENTINA



K64

COMPUTACION PARA TODOS

Dos Proyectos de Interfases

Software Educativo

Los Punteros de la C 64

13 Programas Inéditos

Suplemento de Múltiples Aplicaciones

para CZ 1000/2000, TK 83/85, TI 99 y MSX

SUMARIO

BEGINNERS

Para los que recién se inician, les explicamos cómo descifrar los crípticos mensajes de error que envían las CZ 1000/2000, las TK 83/90X o las TS 2068. Además comentamos cómo cuidar los equipos.

PAG. 14/16

DESARROLLOS

Una interfase paralelo para las CZ 1000 y TK 83 nos permitirá múltiples aplicaciones, mientras que otro proyecto para las Spectrum/TK 90X/TS 2068 será de gran utilidad en caso de tener que tomar datos externos (como temperatura, humedad o iluminación).

PAG. 27

SOFTWARE MEDICO

Un profesional nos relató cómo utiliza una home computer en el servicio de terapia intensiva de un hospital.

PAG. 49

LOS PUNTEROS DE LA C-64



Indicamos en qué consiste esta serie de direcciones de memoria y cómo modificarlas para crear nuevos comandos.

PAG. 52

CARTA DEL DIRECTOR

Como nuestro nombre lo indica, queremos que sea una realidad la "Computación para todos". Por eso más allá de las pocas notas técnicas que incluímos, nos interesa presentar las aplicaciones prácticas de los equipos, tanto en lo que se refiere a las máquinas como a los programas. Claro que algunos proyectos de hardware requerirán ciertos conocimientos (de Basic o de circuítos, por ejemplo), pero como línea general continuamos apuntando a las necesidades de todos los usuarios que quieren aprovechar bien sus computadoras. Para tal fin incluímos desarrollos de múltiples usos para las del tipo CZ 1000 ó TK 83 (aparte de lo que ofrecemos para las 2000 ó 85), software educativo, y consejos útiles para los que recién se inician. Esperemos que les sirvan.

CRISTIAN PUSSO

PROGRAMAS INEDITOS

**TS 1000/1500;
CZ 1000/1500; TK 83/85**

- Bomba (pag. 12)
- Bugy (pag. 12)
- Viaje peligroso (pag. 18)
- Interfase control para CZ 1000/1500, TK 83/85 (pag. 30)

SPECTRUM, TS 2068 y TK 90X

- Cargador de código de máquina (pag. 24)
- Países, capitales y banderas (pág. 34)

TI 99/4A

- Huesos (pag. 38)
- Una subrutina de ploteo (pag. 50)

DREAN COMMODORE 64

- Modificación de los punteros (pag. 52)
- Pelota al blanco (pag. 54)
- Cambio de identificación de disco (pag. 56)

MSX

- Reloj (pág. 40)



Nº 13 ABRIL DE 1986

Director General
Ernesto del Castillo
Director Editorial
Cristian Pusso
Director Periodístico
Fernando Flores

Director Financiero
Javier Campos Malbrán
Coordinador
M.G. Verdomar Weiss
Secretaria
Moni Ocampo

Diagramación
Fernando Amengual
Departamento de Avisos
Oscar Devoto

Departamento de Publicidad
Jefe: Dolores Urien
Promotora:
Mónica Garibaldi

K-64 es una Revista mensual editada por Editorial PROEDI S.A. Cerrito 1320, 1º Piso, Buenos Aires, Te.: 42-9681/9. Registro Nacional de la Propiedad Intelectual: 313.837 M. registrada. Queda hecho el depósito que indica la Ley 11.723 de Propiedad Intelectual.
Impresión: Calcotam. Fotocromo tapa: Columbia. Fotocomposición: Van Waveren.
Distribuidor en Capital: MARTINO, Juan de Garay 358, P.B. Capital. T.E. 361-6962 Distribuidor interior: DGP, Hipólito Yrigoyen 1450, Capital T.E. 38-9266/9800. Los ejemplares atrasados se venderán al precio del último número en circulación.

CORREO ARGENTINO CASA CENTRAL
FRANQUEO PAGO CONCESION Nº 2538
TARIFA REDUCIDA CONCESION Nº 836

Red de distribución

Durante los últimos años los usuarios de microcomputadoras, en cualquiera de sus configuraciones y niveles, han podido observar como, en el exterior del país, la tecnología permitió el desarrollo de bancos de datos a los que similares usuarios de otras latitudes pueden acceder con sólo una llamada telefónica.
En nuestro país Sistronic anunció una red que quedará configurada, en su primera etapa, por líneas conmutadas comunes, los usuarios con sus microcomputadoras y el Host computer de esa empresa, en donde residirán la información que cada usuario podrá consultar y un área de memoria disponible para dejar mensajes entre usuarios (abonados al sistema).
En principio, el software de comunicaciones que deberá poseer el usuario, le será entregado al momento de la inscripción, junto con la descripción del uso, en cassette o diskette a su elección (obviamente, acorde con la configuración de microcomputador de que disponga) y su número clave de abonado. Será necesario que el usuario disponga de un Modem (de por lo menos 300 b.p.s.) el que podrá ser provisto por SISTRONIC.
SISTRONIC permite la resolución de los, hasta hoy, problemas que no permitían la comunicación de equipos micro a macro y viceversa.

Los protocolos de comunicación, sin llevar a los micro a transmisiones de tipo sincrónico, limitaba enormemente el aprovechamiento de recursos de lado del Host, haciendo las aplicaciones muy pesadas y antieconómicas. La resolución siempre estribada en aumentar el costo de los microcomputadores (plaquetas adicionales, unidades de disco para conversión por Soft), o realizar aplicaciones menos rentables, para el lado del Host, o menos atractivas para el usuario.

En los microcomputadores, orientados a comunicación asincrónica, puede obtenerse fácilmente velocidades de transmisión de hasta 9.600 b.p.s. Lamentablemente, el estado de la mayoría de las líneas disponibles en nuestro país, no permite llegar a esa velocidad, pero sí a 1.200 b.p.s. con seguridad.

El equipamiento disponible, permitirá que la velocidad de transmisión comience en alta y vaya bajando, de ser necesario, automáticamente hasta lograr la óptima de acuerdo al estado de línea.

De esta forma, se abre para los usuarios inquietos en desarrollo, la oportunidad de experimentar en comunicaciones, pues estará disponible a ellos la nómina, número de Teléfono, código (de abonado al Sistema) y configuración de su equipo microcomputador, de otros abonados al Sistema. Será posible establecer entonces comunicaciones directas entre abonados.

El QL ataca de nuevo

Sir Clive Sinclair nunca se rinde. Tras los recientes problemas habidos con su salvador, Maxwell, acaba de anunciar increíbles bajas en los precios de sus artículos.

Por ejemplo, el QL se ha comenzado a vender en Inglaterra al irrisorio precio de doscientas libras esterlinas, es decir la mitad de su anterior precio. En algunos comercios británicos se venderá un lote compuesto por un Spectrum Plus, diez programas de aplicación, un grabador de cassettes, una Interfase 2 y un joystick por sólo ciento cuarenta libras.

La operación con la empresa distribuidora Dixons absorberá un stock sobrante de ciento sesenta mil máquinas.

Por otra parte, el equipo científico de desarrollo de Sir Clive continúa trabajando en un nuevo superchip que, según dicen, podrá revolucionar el mundo de las Home Computers. Según informaciones suministradas por Sinclair Research, los principales inconvenientes tecnológicos ya han sido solucionados y el proyecto marcha viento en popa.

Bienvenido cometa "Halley"

El Observatorio Astronómico y Planetario Municipal de la ciudad de Rosario y COMPUTATIONAL-3 S.A. presentan, en una conjunción de esfuerzos, este interesante programa para la CZ-2000 Spectrum.
"HALLEY" es el primero de una serie de programas informativos sobre Astronomía, cuyos desarrollos están en marcha y se dispondrán de ellos durante el transcurso de este año.

Sinclair 128

Mientras que en España ya se lanzó comercialmente, en Inglaterra la Spectrum 128 sigue en los preliminares. El nuevo modelo de Sinclair es compatible en soft con su predecesor, la Spectrum Plus, y tiene características similares en cuanto a su presentación y teclado.
Lo que todavía queda por verse es si aparece software capaz de aprovechar este incremento de memoria. Se piensa vender la 128 con un programa que aprovecha las características de esta máquina.
Este podría ser "The neverending story" (La historia sin fin) de la firma Ocen. Este soft ya existe para Spectrum, pero dada su extensión debe ser dividido en tres partes. En la nueva 128 esto ya no sería necesario.
Como ya sabemos la 128 tiene un Basic mejorado, el chip de sonido AY3-8910 y una mejor sección de video. Mientras tanto, al bajar el precio de la QL a 200 libras, el valor de 140 libras por la Spectrum plus resulta comparativamente alto. Se espera también la Pandora, un modelo portátil de Sinclair y la Sinclair QL II, que tendrá 256 ó 512 K de ram y software de Psion incluido en Rom.



Proyecto europeo rivaliza con la MSX

Mientras la MSX comienza a crecer en Europa los gobiernos de varias naciones se han propuesto crear un nuevo standard en materia de computadores personales.
Bajo el nombre de Eureka, el departamento Británico de comercio e industria propuso un proyecto europeo conjunto para el desarrollo de las normas de standard en los campos de información y entretenimiento hogareño.
Un vocero de MSX, Mike Margolis afirmó que no le preocupa que compañías como la Philips o GEC (que hasta el momento estaban produciendo MSX) se pongan a trabajar en el proyecto Eureka. "Es un verdadero estímulo", dijo, "que otras compañías copien el MSX".
Mientras tanto, las empresas Mitsubishi, Sony, Toshiba y JVC continúan con su apoyo al MSX. Toshiba ha reducido el precio de sus computadores HX 10 a 100 libras, que es el nuevo precio de la GoldStar MSX (máquina de origen Coreano). También ha establecido el MSX2 en su máquina HX 23 que tiene un display de 80 columnas, una resolución de 512 por 212, 256 colores y un reloj perpetuo alimentado por baterías.


usuaria '86
Informática para el usuario

LA ODISEA DE LA QUINTA GENERACION

La revolución informática promete cambiar nuestras vidas con las máquinas que razonan,según el trabajo dado a conocer por Edward Feigenbaum y Pamela McCorduck en el libro "La Quinta Generación" (reeditado por Sudamericana-Planeta) del que publicamos algunos comentarios.

Para la mayoría de las personas los ordenadores son actualmente como un apéndice: nada en lo que merezca la pena pensar excepto cuando nos dan problemas. Son pocas las personas que repiten los tópicos sobre la sociedad informatizada —la posibilidad de que nos convierta a todos en cifras (o en máquinas o en robots)— principalmente porque las cosas no siguieron este camino. En cambio, una encuesta reciente de Harris demuestra que el 60 por ciento de los norteamericanos opina que el ordenador ha mejorado en conjunto la calidad de la vida. Pero aunque el uso del ordenador beneficie a la mayoría de las personas, esto no significa que sea necesariamente atractivo este uso.

Además, los ordenadores son en su mayor parte remotos, abstractos e intangibles. Es difícil imaginar que una crisis de ordenadores afecte realmente a nuestras vidas como la gran escasez de petróleo de 1974, o la repercusión que tiene la escasez local de agua en el consumo del elemento.

¿Qué sucederá cuando una máquina pueda hacer inferencias de modo más profundo que nosotros?

No obstante, la eliminación de los ordenadores de nuestras vidas tendría severas consecuencias. No es fácil darse cuenta, porque nuestros periódicos, compuestos por ordenador, continúan llegando a nuestras puertas con la misma forma de siempre; nuestras revistas mensuales —maravillas de tecnología de satélite controlada por ordenador - continúan siendo entregadas por correo, aunque todavía escribimos nuestros cheques personales, todavía imprimimos acciones con dibujos de fantasía representando inversiones que son simples parpadeos en una base de datos. En definitiva, una pantalla de trivialidades nos ocupa la revolución en marcha.

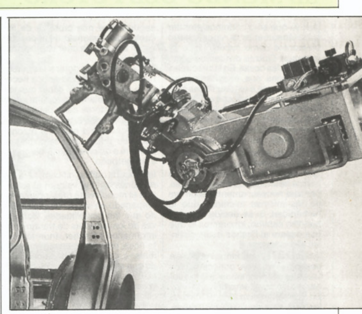
Y continúan formulándose reservas de varios tipos. Hay preocupación por la despersonalización, o por la intimidad, o por cualquier tema del momento. ¿Qué proporción de estas quejas se deben a la tosquedad de una tecnología emergente y nueva? ¿Las reservas que expresa la gente reflejan de verdad su desconcierto ante esta máquina, la menos transparente de todas? Los ordenadores no son de uso tan fácil como deberían ser, incluso hoy día, cuando la interacción humana con ellos ha mejorado mucho en relación al pasado. El comportamiento de los ordenadores está estructurado de modo que parece diferente e incluso extraño a los sistemas humanos de pensar y al lenguaje humano.

A consecuencia de esto, la mayoría de nosotros recurrimos a unos intermediarios entre nosotros y el ordenador, llamados programadores. En esto actuamos como los nobles medievales o los faraones egipcios, que eran analfabetos y tenían que recurrir a los escribas para enviar y recibir mensajes. El personaje no podía saber si su escriba estaba representando sus pensamientos auténticamente, si captaba los matices que él quería introducir, no sabía ni si estos matices eran posibles. Daba sus órdenes y confiaba en que fuesen transmitidas de modo preciso; al otro extremos tenía lugar el proceso inverso, y el personaje primo suyo escuchaba el mensaje. Las oportunidades para el mal eran grandes, porque el poder real estaba en manos de los escribas, los pocos privilegiados que poseían la ciencia de la escritura. Sin duda la tecnología de la escritura resulta extraña e incómoda para el analfabeto, tanto ahora como en el pasado; quizás esto bastaba para que opusiera resistencia contra ella. Sin embargo, ¡qué poder, qué auténtico poder intelectual podían haber tenido aquellos personajes si hubieran escrito ellos mismos sus mensajes!

En la obra de Dickens *La casa inhóspita*, uno de los grandes monumentos que celebran el valor de la información, nos encontramos con Jo, un deshollinador analfabeto que recorre las calles de Londres ignorando totalmente «el significado de aquellos símbolos misteriosos, tan abundantes sobre las tiendas y las esquinas, y sobre las puertas y en las ventanas. ¡Ver leer a la gente, ver escribir a la gente, ver que el cartero entrega las cartas, y no tener la menor idea de todo este lenguaje..., ser completamente ciego y sordo a sus más pequeños elementos! Debe de ser muy intrigante... pensar (porque quizá Jo piensa de vez en cuando) en el posible significado de todo esto. Y si tiene algún significado para alguien, ¿a qué se debe que no signifique nada para mí?»

Lo mismo les sucede a muchos en sus realaciones con ordenadores. Utilizamos el sentido de la *alfabetización* en su acepción más amplia, reconociendo como es lógico que el

término comprende muchas gradaciones: algunas personas consiguen leer un anuncio, pero no una novela de suspenso popular; algunas personas pueden leer una carta comercial, pero no pueden componer una; algunos escritores de poesía y de prosa utilizan el lenguaje como un instrumento, arrancándole composiciones que conmueven y satisfacen la imaginación humana en su nivel más profundo.
Por tanto, el problema de reconciliar las estructuras «naturales» del pensamiento con la tecnología del momento no es nada nuevo. Olvidamos lo difícil que es aprender a leer, y que muchas personas incluso actualmente no lo consiguen. Quizá si los niños aprendieran a usar los ordenadores primitivos de hoy día al mismo tiempo que aprenden a leer —como ya hacen algunos— manejar un ordenador no parecería más raro que leer.
La capacidad de leer y escribir nos ha proporcionado poder, el acceso a un mundo mental opulento e inmenso, ha sido una alteración de los procesos mentales que el analfabeto tiene fuera de su alcance. La capacidad de utilizar un ordenador, incluso en su forma actual, nos abre otro mundo, un mundo en que quizá todos acabarán entrando tan normalmente como entran en el mundo de las letras, y que quizá conferirá un poder superior al que nos han proporcionado ya las poderosas pluma e imprenta. No es una promoción vana. La fuerza muscular del hombre ha sido amplificada por muchas máquinas especializadas y también la fuerza mental del hombre será amplificada. El ordenador no sólo cambiará lo que pensamos, sino cómo pensamos. La red informática que describimos a continuación es un



ejemplo modesto y temprano de este futuro.

Luces y sombras

Ningún cambio profundo en la suerte del hombre ha sido completamente benigno. Incluso la revolución agrícola tuvo algunos efectos secundarios no deseados, aunque pocas personas desearían volver a la caza y a la recolección, con todo lo que implican. En una época mucho más reciente, la amplia distribución de la medicina ha superado nuestra capacidad o nuestra voluntad de controlar la población del mundo, pero tanto la ética como la compasión nos dicen que no deberíamos retirar la medicina, sino prevenir el exceso de población del globo. El gran aumento de saber —en órdenes de magnitud, como hemos señalado insistentemente en esta obra— sin duda no será una excepción. Es de suponer que algunas personas creerán que vivían mejor en los buenos días de antaño.
La creación automática del saber tiene efectos impredecibles. ¿Qué sucederá cuando una máquina pueda utilizar todo el saber que le hemos proporcionado, explotándolo de un modo tan sistemático que supere nuestras capacidades, cuando pueda hacer inferencias de modo más profundo que nosotros (porque no está limitada como nosotros por nuestro legado evolu-

tivo, ya que sólo podemos atender simultáneamente a unos cuatro elementos como máximo)? Lo ignoramos. Quizás olvidemos cómo se hacen las cosas. En las escuelas secundarias nos enseñaron despiadadamente a extraer raíces cuadradas, pero muy pocos adultos recuerdan cómo hay que hacerlo. Las calculadoras portátiles las extraen sin problemas. ¿Por qué cargar nuestra mente con ello?

Ignoramos si un sistema que piense de modo más rápido y profundo que nosotros, aunque utilice la misma heurística que las personas, pensará necesariamente recorriendo las mismas avenidas que la humanidad. Si se decide a seguir por otro camino, ignoramos lo que nos espera al final de estas distintas direcciones.

Ignoramos si una máquina puede descubrir nuevos conocimientos (aunque sospechamos que sí y tenemos indicios tempranos de ello). En caso afirmativo ignoramos cuáles serían las implicaciones de estos nuevos conocimientos.

Ignoramos si una tal red de conocimientos, de alcance mundial, como la imaginan los japoneses, o limitada a una nación, ofrecerá oportunidades sin precedentes para el mal en manos de los gobiernos o de los delincuentes. Nos hemos acostumbrado en épocas de transición como las actuales a que las personas renuncien a sus responsabilidades personales y culpen de ello al ordenador. ¿Serán estas posibilidades cada vez más amenazadoras? ¿Se idearán sistemas legales que puedan resolverlas y que también resuelvan otros problemas como los de los derechos de propiedad intelectual, los de la intimidad y otros problemas impredecibles? ¿Se inventarán sistemas seguros que nos protejan de las cantidades inmensas de poder que están a punto de caer en nuestras manos?

Ignoramos cómo dotar a las personas de la inteligencia crítica necesaria para evaluar el saber que les llega. El problema ya resulta difícil para los lectores de la palabra escrita. Ignoramos si la capacidad de interrogar a una máquina razonadora, de hacer que se explique, ayudará a resolver este problema o a exacerbarlo.

Ignoramos qué significará, para las personas que no valoran el saber, que el mundo esté profundamente empapado en saber. Se ha señalado que las posibilidades recreativas de los sistemas de tratamiento informático del saber, enormemente ricas, permitirán sosegar o estimular a este grupo desemancipado que ahora ridiculiza el saber. No nos atrae especialmente el saber como narcótico, pero la otra posibilidad, que los sistemas de tratamiento informático del saber actúen como estimulantes para saber más, es esperanzadora. Se están diseñando los sistemas antes citados para que puedan utilizarse con tanta facilidad como un teléfono o como el televisor, y podemos recordar como dato optimista que en Estados Unidos el número de televisores pasó de 6.000 a 15,5 millones en cinco años. Confiemos que los sistemas de tratamiento informático del saber tengan un éxito igual.

No hace mucho, Feigenbaum estaba en el aeropuerto de San José a punto de subir a un avión. Llegó un aparato antiguo, un bello biplano que había sido uno de los primeros aviones de pasajeros de Trans World Airlines. Se le ocurrió que la ingeniería del saber y los sistemas especializados estaban precisamente en esta fase; estaban luchando por transformarse y pasar de ser una novedad tecnológica potencialmente potente a formar parte integrante de la vida humana. Los aeroplanos todavía no son perfectos. A veces llegan tarde; a veces se estrellan catastróficamente. Pero son nuestros y apenas podríamos imaginarnos la vida sin ellos. Las máquinas de inferencia simbólica están en el mismo estadio que el avión de Trans World que vio Feigenbaum: una promesa elegante de futuro.

Sin embargo, debemos volver a un presente más sombrío. Hemos descrito en este libro una tecnología que promete cambiar nuestras vidas como pocas lo han hecho: las máquinas razonadoras no son simplemente, como ya hemos dicho, la segunda revolución informática, sino la revolución importante. Si los detalles de la tecnología en sí son complicados, los temas que la rodean son comprensibles para casi todo el mundo. Una superioridad en la tecnología del saber proporciona a quien la domina el poder para resolver los matices de gris y transformarlos en blanco y negro, proporciona en definitiva una ventaja inequívoca, tanto si hablamos de poder personal como de economía nacional o de poder bélico.

Los japoneses lo han entendido perfectamente. Han empezado ya a traducir esta idea en una nueva tecnología que les proporcionará una ventaja inequívoca sobre el resto del mundo, quizás a mediados de la próxima década. Otras naciones reconocen que la estrategia japonesa es lógica, y que desde luego es inevitable. Se están elaborando en muchos lugares ambiciosos planes nacionales.



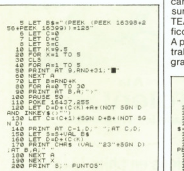
PROGRAMAS

BOMBA





La idea de "bomba" es la de destruir una cantidad indefinida de enemigos, representados por un espacio invertido. Las bombas caerán desde nuestra nave cuando pulsemos cualquier tecla. Hay que tener en cuenta que no se puede lanzar una nueva bomba si no ha desaparecido aún la anterior de la pantalla.
Para usar este programa en modo slow, debemos borrar las instrucciones pause y poke, de las líneas 100 y 110.



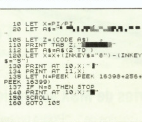
```
   5 LET B$="(PEEK (PEEK 16398+2
56*PEEK 16399))=128"
   6 LET C=0
   7 LET D=C
   8 LET S=C
  10 LET K=9.5
  20 FOR X=1 TO 5
  30 CLS
  40 FOR A=1 TO 5
  50 PRINT AT 9,RND*31;"■"
  60 NEXT A
  70 LET B=RND*K
  80 FOR A=0 TO 30
  90 PRINT AT B,A;">"
 100 PAUSE 50
 110 POKE 16437,255
 120 LET D=D+(C<K)+A*(NOT SGN D
AND INKEY$<>"")
 130 LET C=(C+1)*SGN D+B*(NOT SG
N D)
 140 PRINT AT C-1,D;" ";AT C,D;
 150 LET S=S+VAL B$
 160 LET D=D+(C<K)
 170 PRINT CHR$ (VAL "23"*SGN D)
;AT B,A;" "
 180 NEXT A
 190 NEXT X
 200 PRINT S;" PUNTOS"
```

BUGY

COMP.: CZ 1000/1500; TK 83/85
CONF.: 2 K
CLAS.: ENT



Basado en un juego de Arcade, este es un programa que produce gran adicción por ser muy entretenido.
El objetivo de éste es deslizar el bugy a lo largo de la pista usando las teclas 5 y 8, para moverlo a la izquierda y a la derecha respectivamente.
La pista esta definida en la línea 20, dentro de la variable a$, los caracteres gráficos de dicha línea surgen de pulsar las teclas 127845 TEADSDAET54721 en modo gráfico.
A pesar de estar hecho en basic y trabajar en modo slow, este programa es bastante rápido.



```
  10 LET X=PI/PI
  20 LET A$="[illegible]"
 105 LET Z=(CODE A$)
 110 PRINT TAB Z;"[illegible]"
 112 LET A$=A$(2 TO )
 120 LET X=X+(INKEY$="8")-(INKEY
$="5")
 130 PRINT AT 10,X;"■"
 134 PRINT AT 11,X;
 135 LET N=PEEK (PEEK 16398+256*
PEEK 16399)
 137 IF N=8 THEN STOP
 140 PRINT AT 10,X;"■"
 150 SCROLL
 160 GOTO 105
```

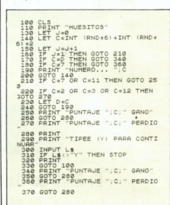
HUESITOS

COMP.: CZ 1000/1500; TK 83/85
CONF.: 2 K
CLAS.: ENT

Si el primer número que saque en la tirada es 7 u 11 ganaremos automáticamente.
Un puntaje de 2, 3 ó 12 en su primer intento hará que perdamos sin remedio.
Cualquier otro número nos dará la oportunidad de un nuevo tiro.

Estructura del programa

La tirada de dados es simulada por medio de la instrucción RND.
Una forma simple de utilizar esta función para generar números entre 1 y 7 es la siguiente:
LET C=INT (RND*6)+1
y para hacerlo entre 1 y 12 el programa usa la siguiente fórmula:
LET C=INT(RND*6)+INT (RND*6+2)



```
 100 CLS
 110 PRINT "HUESITOS"
 130 LET J=0
 140 LET C=INT (RND*6)+INT (RND*
6)+2
 150 LET J=J+1
 160 IF J=1 THEN GOTO 210
 170 IF C=D THEN GOTO 340
 180 IF C=7 THEN GOTO 360
 190 PRINT "NUMERO... ";C
 200 GOTO 140
 210 IF C=7 OR C=11 THEN GOTO 25
0
 220 IF C=2 OR C=3 OR C=12 THEN
GOTO 270
 230 LET D=C
 240 GOTO 190
 250 PRINT "PUNTAJE ";C;" GANO"
 260 GOTO 280
 270 PRINT "PUNTAJE ";C;" PERDIO
"
 280 PRINT
 290 PRINT "TIPEE (Y) PARA CONTI
NUAR"
 300 INPUT L$
 310 IF L$<>"Y" THEN STOP
 320 PRINT
 330 GOTO 100
 340 PRINT "PUNTAJE ";C;" GANO"
 350 GOTO 280
 360 PRINT "PUNTAJE ";C;" PERDIO
"
 370 GOTO 280
```




LA CAZA DE ERRORES

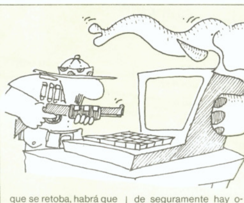
Con el rastreo y corrección de las equivocaciones aprendemos cómo funciona un programa. Esta nota nos ayuda a descifrar los crípticos mensajes que nos envía la máquina.

**SPECTRUM / 2068
TK90X / TK83/85
CZ1000/1500**

El tipeado de un programa es un ejercicio muy útil. Aparte de la paciencia requerida, las técnicas aprendidas y el programa terminado, probablemente con lo que más se aprende es con el rastreo y correción de errores (léase "debugging"), ya sea por propia culpa o por problemas de impresión inevitables...

La caza de errores nos acerca al funcionamiento del programa en sí, ya que no hay más remedio que estudiarlo para deducir qué es lo correcto. Resulta en un arte de por sí, y se necesita de cierto "adiestramiento" y habilidad para salir airoso.

Aquí van algunas recetas que tal vez ayuden en sus esfuerzos para resolver uno de esos crípticos mensajes de error.



1 NEXT without FOR

Revisar hacia atrás el programa. O el bucle no está aún establecido, o no está relacionado con una línea: FOR a=N1 TO N2, o la variable ya ha sido reutilizada como variable ordinaria dentro del bucle mismo en un comando LET.

2 Variable not found

Es uno de los errores más frecuentes. Una vez más, puede que el error no esté necesariamente en la línea donde el error fue detectado y reportado. Puede que, si el error está en esa línea, ésta tenga varias variables en ella. Para pescar cuál es la que se retoba, habrá que probar a mano en ese momento una por una. Supongamos que tenemos: PRINT AT A,B; S$

Probaremos;

PRINT A

PRINT B

PRINT S$

Hasta hallar la que provoca el error. A continuación, revisar el programa hacia atrás, en búsqueda de la línea donde se define esa variable. Seguramente se la comió o apareció un GOSUB o GOTO que lo desviaba y nunca quedaba definida.

3 Suscript wrong

Está conectado con DIM A (N) o DIM A$ (N). Si el número entre paréntesis de la línea que produjo el reporte es mayor que el original definido en la línea del DIM, no es entero o es una letra, entonces se genera ese informe de error.

Si el número entre paréntesis es un número, chequearlo y arreglarlo. Si es una letra, estudiar cómo se define ese valor.

Seguramente excedió los límites, ver las líneas donde se ajusta su valor, donde seguramente hay operadores del tipo + - * /. Si es necesario, fijar los límites de alguna manera, por ej;

IF X>10 THEN LET X=10

4 Out of memory

Puede ocurrir tanto en programas muy largos para la memoria de la máquina, como también si previamente se corrió el RAMTOP.

Antes de desesperar, entrar CLEAR USR "a"-1 en la Spectrum. En las 1000, Savear el programa; apagar y prender y volver a cargar el programa.

7 RETURN without GOSUB

De algún modo, la máquina encontró un RETURN sin venir de un comando GOSUB. Verificar si se equivocó y puso un GOTO en vez de un GOSUB o si sencillamente se olvidó de ponerlo.

B Integer out of range

Es el caso cuando aparece un número entero (sin parte fraccionaria), y resulta muy chico o muy grande para utilizarlo para alguna tarea como: PRINT AT 0,35 que no se puede!

Chequear variables como en el caso de error 2. En todo caso agregarle límites como en caso del error 3.

E Out of DATA

Es un problema de Spectrum. Chequear la cantidad de números en la sentencia DATA, que tiene que coincidir con los de la READ.

Por lo general uno se come por lo menos uno o dos. Si se comió más, revisar la vista o el apetito! Lo mismo pasa si se trata de releer una lista DATA sin antes ejecutar un RESTORE: Suele pasar en programas de ejecución automática (saveados con LINE) la buena práctica de programación hace que se RESTORE en la línea correcta antes de usar el READ.

I FOR without NEXT

Ver lo mismo que en el error 1, pero acá se perdió el NEXT.

Notar que las letras usadas en los ejemplos pueden ser cualquiera.

Esta lista no trata por supuesto de ser completa, pero creemos que ayuda bastante. Sólo trata de ser una continuación de otras recetas dadas anteriormente.

Resulta gratificante cada vez que nos vemos sumergidos en la caza de errores, cuando nos ponemos a teclear un programa. Pero queremos dejar por sentado acá, que de ninguna manera son ciertos esos rumores que dicen por ahí que en K64 introducimos errores deliberadamente en nuestros listados para iniciar a nuestros lectores en estos dudosos placeres del debugging!.

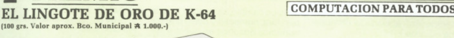
SELECCIONAMOS

EL PROGRAMADOR DEL AÑO '86

K64

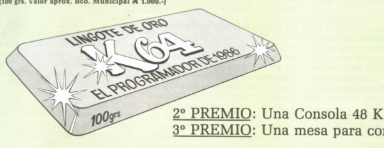
COMPUTACION PARA TODOS

1º PREMIO



EL LINGOTE DE ORO DE K-64

(100 grs. Valor aprox. Bco. Municipal ₳ 1.000.-)



2º PREMIO: Una Consola 48 K

3º PREMIO: Una mesa para computador

BASES PARA PARTICIPAR EN EL CERTAMEN

Las bases y condiciones generales son las siguientes:

Una vez terminado y revisado tu programa, deberás enviarlo a la editorial grabado en un cassette o diskette, varias veces para mayor seguridad. (Inclusive grabado con dos grabadores distintos). Indicar en el cassette o diskette, los datos del programa, computadora y autor.

Otra condición es que sea original e inédito, es decir que no haya sido enviado a ninguna otra publicación. Si bien es preferible que vaya acompañado del listado del mismo por impresora, este no es imprescindible.

El programa deberá venir con un texto que aclare cuál es su nombre, objetivo, modo de uso, y explicación de cada una de sus partes, subrutinas y variables. Si posee lenguaje de máquina, es fundamental una buena explicación sobre su funcionamiento e ingreso a la máquina. No olvidarse los datos completos del autor o autores.

El texto se presentará en hojas tipo oficio y mecanografiado a doble espacio. No importa que la redacción no sea muy clara, eso queda por nuestra cuenta.

Jurado: Un jurado propio compuesto por profesionales en computación y usuarios de computadores, decidirá los resultados del mismo.

El criterio de elección, como siempre, se basará en originalidad de la idea; método de programación; efectos gráficos y sonoros; documentación del programa; presentación y ahorro de memoria. En la clase de programas del tipo no-juegos, se evaluará también la facilidad de manejo y explicación de los conceptos vertidos en el programa.

Cierre: El cierre de recepción de trabajos para concurso de programas será el: 31/07/86. (K64 se reserva el derecho de publicación de los programas recibidos, como asimismo la devolución del material).

Sorteo Mensual: Todos los meses se sortearán 20 cassettes entre los programas recibidos.



CUIDEMOS EL EQUIPO

El mantenimiento de las microcomputadoras es muy sencillo y toma poco tiempo. El cuidado de unos pocos detalles evita fallas intempestivas.

Un sistema de computación descuidado trae normalmente unos cuantos problemas. Entre ellos, una pantalla borrosa debido a suciedad o TV descalibrado, diskettes y cassettes imposibles de cargar por estar en mal estado los decks, impresión deficiente de la impresora, etcétera.

Para tener todo en orden sólo bastan unos minutos por semana de "dedicación" al cuidado de nuestros equipos.

Respecto al tema de la limpieza, basta con usar los elementos comunes para mantener bien transparente la pantalla de nuestro TV o monitor. Los cabezales de grabadores y su ruedita de goma se limpian con alcohol o tetracloruro de carbono con un isopo. Los cassettes limpiacabezales no dan tan buen resultado.

Para las disketteras, se consiguen discos limpiadores especiales para cada drive.

Las impresoras por su parte, no tienen en este sentido grandes secretos. Hay que retirar bien el polvillo que se forma dentro con un trapo apenas humedecido. De paso observar el estado de la cinta, el cabezal y las ruedas tractoras. ¡Seguramente están pidiendo a gritos un poco de atención!

Antes las computadoras (aunque hoy en día algunas también), se escondían en ambientes especiales con aire filtrado y lejos de cualquier disturbio. Nuestras pobres maqui-



nitas en cambio, se bancan la temperatura, humedad y polvo que soportemos nosotros en nuestra habitación.

La "buena práctica" recomienda limitar estos efectos nocivos en lo posible, como así también en aquellos casos en que la falta de lugar es grave y se la pasan armando y desarmando el conjunto continuamente. Esto desgasta muy rápidamente los conectores de máquina y periféricos con consecuencias de lo más nefastas. Una mesa dedicada al equipo es lo ideal, donde estén ordenados todos los cables y accesorios. Por lo menos, que lo único que se esté colocando y sacando sea el televisor "familiar".

Para cuidarse del polvo, es ideal cubrir todo con fundas a medida para cada máquina y periférico. Hay algunas que al ser translúcidas, permiten ver si uno se olvidó la máquina encendida (la C64 por ejemplo). El polvo en los intersticios queda horrible!, sobre todo entre las teclas y lugares por el estilo, que es difícil llegar para limpiar. Recordemos que el polvo metido en los contactos de las teclas, dificulta su funcionamiento. Además de la necesidad técnica de la limpieza, el buen aspecto y cuidado de nuestra computadora la valorizará mucho más a la hora de querer venderla.

Otros enemigos de las computadoras son el cigarrillo, sus ceniceros cercanos, las aspiradoras funcionando y las bebidas susceptibles de ser derramadas por manos temblorosas.

Siguiendo con los cucos de las micros, está el de la electricidad estática, que en ambientes y días muy secos pueden ser peligrosos para los chips. Esta electricidad suele acumularse mucho sobre la pantalla del televisor, sobre todo en los de gran tamaño y a color. Estos televisores además traen problemas en la carga de cassettes y diskettes al captar los drives la radiación interferente que producen aquéllos, proveniente del transformador de barrido horizontal.

Los demás accesorios sólo requieren una revisión periódica de sus cables y conectores, que no estén flojos o desligados. Por otro lado, el tener los cassettes y diskettes bien rotulados y ordenados, es bueno, ya que no se sabe cuando necesitaremos aquel programita perdido...

En fin, el mantenimiento preventivo no es sólo una cuestión de estética y puede constituir la diferencia entre un equipo que funcione bien y otro que nos vuelva locos.



C U R S O S

VIAJE PELIGROSO

COMP.: CZ 1000/1500; TK 83/85
CONF.: 16 K
CLAS.: ENT
AUTOR: RICARDO H. ORQUERA
Bs. As.



Mapa 5	Mapa 6
Mapa 3	Mapa 4
Mapa 1	Mapa 2

Figura 1.

El mismo se inspira en el filme "El motín del Boutny" del cual existen por lo menos dos versiones, siendo la primera protagonizada nada menos que por Marlon Brando y, en cuanto a la segunda de reciente estreno, todavía no he tenido el placer de verla. Bueno, yendo al grano, la idea era intentar desarrollar íntegramente en Basic y con una programación lo más estructurada posible, (dentro de lo que permite el Basic de la TS-1000 para la que fue hecho) un entretenimiento que incluye las dosis necesarias de aventura y presentación gráfica, como para mantener la mayor "adicción" posible del usuario. Para la presentación gráfica la pantalla fue dividida en dos ventanas y un display de estado.

En la parte superior de la misma y en forma longitudinal se ubica la ventana de mensajes, este área es utilizada por el programa para comunicar diversos mensajes sobre las condiciones de navegación y otros eventos como ya veremos.

En la ventana inferior se muestran los distintos mapas de navegación y la zona izquierda de la parte inferior es continuamente actualizada con las variables que hacen al desarrollo del juego. En el mismo el usuario es el capitán del buque velero "HMS Boutny", el que lo transportará por peligrosos mares a través de esta aventura. Cuando el viaje comienza y luego de una pantalla donde se presentan sintéticamente estas instrucciones el jugador se encuentra ubicado en la esquina inferior izquierda del mapa Nº 1 y su misión será arribar a puerto de destino en la esquina superior derecha del mapa Nº 6 (ver figura 1) con la mayor cantidad posible de barras de oro y en el menor tiempo. Para ello el jugador deberá en principio, tomar una serie de decisiones y enfrentar la posibilidad de ciertos peligros. En primer lugar, será informado en la ventana superior en cuanto a la intensidad y dirección del viento, sobre esta base deberá decidir qué superficie de velamen desplegará (dentro de un cierto máximo que varía según veremos) y la dirección o curso; la superficie se mide en metros cuadrados, y el curso en grados, a partir de 000 grados norte y en el sentido de las agujas del reloj. De acuerdo a la superficie desplegada y a la dirección e intensidad del viento nuestra nave se desplazará una cierta cantidad de espacios dentro del mapa en la dirección elegida siempre y cuando la misma se aparte en, por lo menos, 45º de la dirección frontal del viento, de lo contrario el buque no se moverá, habiendo perdido una jornada. Por cada movimiento pasará una jornada a partir del día 0 ó de partida, y por cada jornada nuestra tripulación (en principio 100 hombres) consumirá una ración por tripulante (empezamos con 2.500 raciones). En nuestra travesía encontraremos sitios marcados con las letras "A" a la "Z" en video inverso, éstos son los lugares donde se hayan enterrados los tesoros consistentes en barras de oro, y arribando a ellos, el oro encontrado pasará a engrosar nuestras arcas (al comienzo poseemos 1.000 barras).

Cuadro de explicaciones

Cómo juzgar su desempeño

B/Oro- Performance
1000<=Valió la pena el esfuerzo
1000-2500: Apenas alcanza para pagar sueldos atrasados.
2500-5000: Exito moderado.
5000-7500: Bien hecho, obtuvo buenas ganancias.
>=7500: Hizo una verdadera fortuna y logró su promoción a almirante.

Dificultades

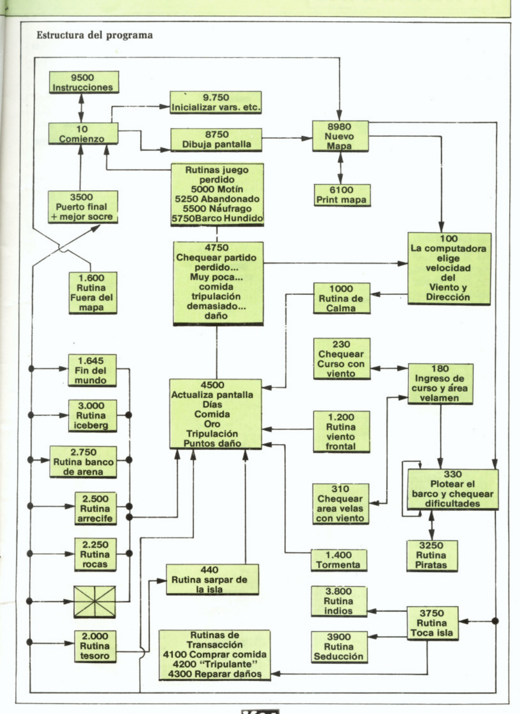
Rocas: Su barco será dañado.
= Arrecife: Su barco será dañado.
Iceberg: Su barco será dañado.
+ Banco de arena: Su barco será dañado.
Isla: Ud. tiene 20% de chance de ser atado por salvajes y 20% de perder tripulantes en manos de las indias, o se le ofrecerá la opción de comprar comida, hombres o repara su buque si su oro alcanza.

Variables principales

O: Oro (puntaje)
OA: Mejor score.
C: Raciones de comida.
FT: Fuerza de tripulación.
E: Estado del buque.
E1: Nivel de averías.
EA: Estado previo del barco.
FV: Fuerza del viento.
DV: Dirección del viento.
AM: Area máxima de velamen manejable por la tripulación disponible.
A: Area de velamen ingresada.
CU: Curso ingresado.
Q(26): Codes CHR$ de los tesoros.
T: Días de navegación.
N: Número del mapa.
D: Distancia que se debe mover la nave.
Z: Flag reimprimir scores.
X,Y: Coordenadas posición buque.
X1,Y1: Coordenadas de la última posición del buque.
X2,Y2: Coordenadas del barco al final del día anterior.
L,K: conversión de X,Y en posiciones PRINT.
L1,K1: Conversión de X1,Y1 en posiciones PRINT.
C$: Array del mapa.
T$: Tipo de operación elegida.
P$,Q$,R$,S$: Mensajes para ahorrar memoria.

Secuencia del programa

LINEA
100: Programa principal.
180: Entrada de curso y área del velamen.
230: Chequea curso con dirección del viento.
310: Chequea área de las velas con fuerza del viento.
330: Plotea barco y chequea lugar.
1000: Rutina de calma.
1200: Rutina de viento en contra.
1400: Rutina de tormenta.
1600: Rutina fuera del mapa.
1645: Rutina fin del mundo.
2000: Rutina tesoro.
2250: Rutina de rocas
2500: Rutina de arrecife.
2750: Rutina banco de arena.
3000: Rutina iceberg.
3250: Rutina piratas.
3500: Rutina puerto final, incluye mejor score.
3750: Rutina isla.
3800: Rutina ataque indios.
3900: Rutina seducción indias.
4000: Rutinas de transacción.
4100: Comprar comida.
4200: Tomar tripulantes.
4300: Reparar averías buque.
4400: Rutina zarpar de la isla.
4500: Rutina actualizar display.
4750: Chequea partido perdido.
5000: Rutina motín.
5250: Rutina náufrago.
5500: Rutina abandonado.
5750: Rutina barco se hunde.
6000: Limpiar área de mensaje.
6100: Mostrar nuevo mapa.
7250: Arrays crear mapas.
8750: Dibujar pantalla.
9500: Instrucciones.
9750: Inicializar variables.



Estructura del programa
9500 Instrucciones
9.750 Inicializar vars. etc.
10 Comienzo
8750 Dibuja pantalla
8980 Nuevo Mapa
Rutinas juego perdido
5000 Motín
5250 Abandonado
5500 Náufrago
5750Barco Hundido
6100 Print mapa
3500 Puerto final + mejor socre
4750 Chequear partido perdido... Muy poca... comida tripulación demasiado... daño
100 La computadora elige velocidad del Viento y Dirección
1.600 Rutina Fuera del mapa
1000 Rutina de Calma
230 Chequear Curso con viento
1.645 Fin del mundo
180 Ingreso de curso y área velamen
4500 Actualiza pantalla Días Comida Oro Tripulación Puntos daño
3.000 Rutina iceberg
1.200 Rutina viento frontal
2.750 Rutina banco de arena
330 Plotear el barco y chequear dificultades
2.500 Rutina arrecife
310 Chequear area velas con viento
2.250 Rutina rocas
1.400 Tormenta
3250 Rutina Piratas
440 Rutina sarpar de la isla
3.800 Rutina indios
3750 Rutina Toca isla
2.000 Rutina tesoro
Rutinas de Transacción
4100 Comprar comida
4200 "Tripulante"
4300 Reparar daños
3900 Rutina Seducción




```
  10 SAVE "BOUTNY"
  20 LET OA=0
  30 RAND
  40 GOSUB VAL "9500"
  42 POKE 16418,0
  43 PRINT AT 23,0;" PULSE UNA T
ECLA PARA COMENSAR.."
  44 PAUSE 4E4
  45 PRINT AT 23,1;"POR FAVOR ES
PERE.............."
  46 GOSUB VAL "9750"
  47 POKE 16418,2
  50 GOTO VAL "8750"
 105 GOSUB VAL "6000"
 110 LET FV=INT (RND*8)
 115 IF FV=0 THEN LET FV=INT (RN
D*8)
 120 IF FV=7 THEN LET FV=FV+INT
(RND*5)
 130 IF FV=11 THEN LET FV=FV+INT
 (RND*2)
 140 LET DV=INT (RND*360)
 150 PRINT AT 2,1;"FUE.DELVIENTO
";FV;
 152 LET C=(C-FT)*(C>FT)
 154 LET T=T+1
 160 IF FV=0 THEN GOTO VAL "1000
"
 170 PRINT ";";DV;" GRDS.DELN"
 180 PRINT TAB 1;"RUMBO? ( GRD.
DESDE EL N )"
 190 INPUT CU
 200 IF CU<0 OR CU>359 THEN GOTO
 VAL "190"
 210 PRINT AT 3,7;"
 "
 220 PRINT AT 3,8;CU;" GRADOS DE
SDE EL N"
 230 IF ABS ((CU+360*(CU<45 AND
DV>315))-(DV+360*(CU>315 AND DV<
45)))<45 THEN GOTO VAL "1200"
 240 LET CU=PI*CU/180
 250 LET AM=50*FT
 260 PRINT TAB 1;"AREA VELA? (MA
X. ";AM;" MTS.CDS)"
 270 INPUT A
 280 IF A<0 OR A>AM THEN GOTO VA
L "270"
 290 PRINT AT 4,10;"
 "
 300 PRINT AT 4,11;A;" MTS.CDS.
IZADOS"
 310 IF FV>6 AND A>500*(12-FV) T
HEN GOSUB VAL "1400"
 320 LET D=.00001*A*FV**1.5*E
 330 FOR R=0 TO D STEP 2
 340 LET X=X2+R*SIN CU
 350 LET Y=Y2+R*COS CU
 360 IF X<-.5 OR X>=49.5 OR Y<-.
5 OR Y>=29.5 THEN GOTO VAL "1600
"
 370 UNPLOT X1+12,Y1+2
 380 LET L1=15-INT ((Y1+.5)/2)
 385 LET K1=1+INT ((X1+.5)/2)
 390 LET L=15-INT ((Y+.5)/2)
 395 LET K=1+INT ((X+.5)/2)
 400 PRINT AT 5+L1,5+K1;D$(L1,K1
)
 410 PLOT X+12,Y+2
 420 IF R=0 OR D$(L,K)=" " THEN
GOTO VAL "500"
 430 IF CODE D$(L,K)>165 THEN GO
TO VAL "2000"
 440 IF D$(L,K)="▞" THEN GOTO VA
L "2250"
 450 IF D$(L,K)="=" THEN GOTO VA
L "2500"
 460 IF D$(L,K)="*" THEN GOTO VA
L "2750"
 470 IF D$(L,K)="▀" THEN GOTO VA
L "3000"
 480 IF D$(L,K)="■" THEN GOTO VA
L "3500"
 490 IF D$(L,K)="▒" THEN GOTO VA
L "3750"
 500 IF INT (RND*250+1)=5 THEN G
OSUB VAL "3250"
 510 LET X1=X
 520 LET Y1=Y
 530 NEXT R
 540 LET X2=X
 550 LET Y2=Y
 600 GOTO VAL "4500"
1010 FOR R=1 TO 50
1020 NEXT R
1025 GOSUB VAL "6000"
1030 PRINT AT 2,1;"CALMA UD. DER
IVA INPOTENTE"
1040 PRINT TAB 7;"MIENTRAS SU TR
IPULACION"
1050 PRINT TAB 7;"CONSUME ";FT;S
$
1060 GOTO VAL "4500"
1205 GOSUB VAL "6000"
1210 PRINT AT 2,1;" VIENTO  UD.
NO PUEDE NAVEGAR"
1220 PRINT "▌FRONTAL  CONTRA EL
VIENTO"
1230 PRINT TAB 10;"CONSUME ";FT;
S$
1390 GOTO VAL "4500"
1410 GOSUB VAL "6000"
1420 LET E1=10*(FV-2)
1440 IF FV<12 THEN PRINT AT 2,1;
"TORMENTA  ";
1450 IF FV=12 THEN PRINT AT 2,1;
"HURACAN  ";
1455 PRINT "CUBRA LAS ESCOTILLAS
"
1460 PRINT AT 3,11;O$
1470 PRINT TAB 11;E1*(E>E1)+E*(E
<=E1);R$
1480 LET E=(E-E1)*(E>E1)
1490 RETURN
1605 LET D=D-R
1610 IF (N/2<>INT (N/2)) AND (X>
=47.5) THEN GOTO VAL "1700"
1620 IF (N/2=INT (N/2)) AND (X<1
.5) THEN GOTO VAL "1725"
1630 IF (N<5) AND (Y>=27.5) THEN
 GOTO VAL "1750"
1640 IF (N>2) AND (Y<1.5) THEN G
OTO VAL "1775"
1645 GOSUB VAL "6000"
1650 PRINT AT 2,1;" FIN DEL LA T
RIPULACION LO "
1660 PRINT TAB 1;"  MUNDO  PERSU
ADE A NO CONTI-"
1670 PRINT TAB 10;"NUAR.(TIERRA
PLANA)"
1672 LET X=X1
1674 LET X2=X1
1676 LET Y=Y1
1678 LET Y2=Y1
1680 GOTO VAL "4500"
1700 LET N=N+1
1705 LET Y2=Y1
1710 LET X2=X1-48
1715 LET X1=X2
1720 GOTO VAL "8980"
1725 LET N=N-1
1728 LET Y2=Y1
1730 LET X2=X1+48
1735 LET X1=X2
1745 GOTO VAL "8980"
1750 LET N=N+2
1755 LET X2=X1
1760 LET Y2=Y1-28
1765 LET Y1=Y2
1770 GOTO VAL "8980"
1775 LET N=N-2
1780 LET Y2=Y1+28
1785 LET Y1=Y2
1795 GOTO VAL "8980"
2010 GOSUB VAL "6000"
2020 LET O=O+10*CODE D$(L,K)
2030 PRINT AT 2,1;"TESORO UD. EX
PLORA EN TIERRA."
2040 PRINT TAB 8;"ALLI ENCUENTRA
 UN COFRE"
2050 PRINT TAB 8;"CON ";10*CODE
D$(L,K);" BARRAS DE ORO"
2060 LET Q(CODE D$(L,K)-165)=8
2070 LET D$(L,K)=CHR$ 8
2240 GOTO VAL "4400"
2260 GOSUB VAL "6000"
2262 LET EA=E
2265 LET E=INT (E*(1-RND/2))-2
2270 PRINT AT 2,1;"ROCAS     ";P
$
2280 PRINT TAB 11;Q$
2290 PRINT TAB 11;EA-E;R$
2310 GOTO VAL "4500"
2510 GOSUB VAL "6000"
2512 LET EA=E
2515 LET E=INT (E*(1-RND/4))-2
2520 PRINT AT 2,1;"ARRECIFE  ";P
$
2530 PRINT TAB 11;O$
2540 PRINT TAB 11;EA-E;R$
2560 GOTO VAL "4500"
2760 GOSUB VAL "6000"
2770 PRINT AT 2,1;"BANCO DE ";P$
2780 PRINT "▌ ARENA   BARCO ENCA
LLADO."
2790 PRINT TAB 10;"CONSUME ";FT;
S$
2800 GOTO VAL "4500"
3010 GOSUB VAL "6000"
3012 LET EA=E
3015 LET E=INT (E*(1-RND/2))-2
3020 PRINT AT 2,1;"ICEBERG    ";P
$
3030 PRINT TAB 11;O$
3040 PRINT TAB 11;EA-E;R$
3060 GOTO VAL "4500"
3260 GOSUB VAL "6000"
3270 PRINT AT 2,1;"PIRATAS   ROB
AN TODO SU ORO  "
3272 LET O=0
3276 LET FT1=FT
3278 LET FT=INT ((FT-1)/(2-RND))
3280 PRINT TAB 11;FT1-FT;" HOMBR
ES MUERTOS"
3286 LET E1=INT ((E-1)/(2-RND))
3290 PRINT TAB 11;E-E1;" PUNTOS
DE AVERIA"
3292 LET E=E1
3300 FOR R=1 TO 100
3310 NEXT R
3490 RETURN
3505 LET Z=1
3506 GOSUB VAL "4500"
3510 GOSUB VAL "6000"
3520 PRINT AT 2,1;"PUERTO  NAVE
AMARRADA A SEGURO"
3530 PRINT TAB 1;"FINAL   LUEGO
DE ";T;" DIAS EN"
3540 PRINT TAB 9;"EL MAR CON ";O
;" BARRAS";TAB 12;"[illegible]"
3550 IF O>OA THEN GOSUB VAL "360
0"
3560 GOTO VAL "42"
3600 LET OA=O
3610 FOR R=1 TO 25
3620 PRINT AT 1,12;"MEJOR SCORE"
3630 PRINT AT 1,12;"MEJOR SCORE"
3650 NEXT R
3690 RETURN
3755 LET FT1=FT
3756 GOSUB VAL "6000"
3760 LET R=RND
3770 IF R>=.4 THEN GOTO 4000
3780 IF R>=.2 THEN GOTO 3900
3810 LET FT=INT (FT*(1-RND/2))-2
3830 PRINT AT 2,1;"TIERRA    LOS
 INDIOS ATACAN."
3840 PRINT TAB 11;FT1-FT;" HOMBR
ES MUERTOS."
3850 PRINT TAB 11;"UD. ESCAPO PO
R POCO."
3860 GOTO VAL "4400"
3910 LET FT=INT (FT*(1-RND/4))-2
3920 PRINT AT 2,1;"TIERRA  SEDUC
TORAS DONCELLAS"
3930 PRINT TAB 9;FT1-FT;" HOMBRE
S LO DEJAN."
3940 PRINT TAB 9;"*MUCHACHOS SUE
RTUDOS*"
3950 GOTO VAL "4400"
4005 LET Z=1
4006 GOSUB VAL "4500"
4010 PRINT AT 2,1;"TIERRA DEBE C
OMPRAR O NAVEGAR"
4020 PRINT TAB 8;"COM.(C),TRP.(T
),REP.(R)"
4030 PRINT TAB 8;"NAVEGAR(N)..QU
E OPCION?"
4040 IF INKEY$<>"" THEN GOTO VAL
 "4040"
4042 IF INKEY$="" THEN GOTO VAL
"4042"
4045 LET T$=INKEY$
4048 GOSUB VAL "6000"
4050 IF T$="C" THEN GOTO VAL "41
00"
4055 IF T$="T" THEN GOTO VAL "42
00"
4060 IF T$="R" THEN GOTO VAL "43
00"
4065 IF T$="N" THEN GOTO VAL "44
00"
4070 GOTO VAL "4000"
4075 PRINT AT 3,8;"SU CAPITAL NO
 ALCANSA"
4080 PRINT AT 10,0;"▌    ";AT 10
,1-(O>9999);O
4085 PRINT AT 4,4;"PULSE UNA TEC
LA PARA SEGUIR"
4090 PAUSE 4E4
4092 GOSUB VAL "6010"
4095 GOTO VAL "4010"
4100 PRINT AT 2,1;"COMIDA 1 RACI
ON/1 BARRA DE ORO"
4110 PRINT TAB 8;"CUANTAS RACION
ES?"
4120 INPUT R
4130 IF (O-R)<0 THEN GOTO 4075
4135 IF C+R>2500 THEN PRINT AT 4
,8;"NO HAY LUGAR EN BODEGA"
4136 IF C+R>2500 THEN GOTO 4120
4140 LET C=C+R
4145 PRINT AT 12,1;"    ";AT 12,
1;C
4150 LET O=O-R
4160 PRINT AT 3,8;R;" RACIONES C
OMPRADAS"
4170 GOTO VAL "4080"
4200 PRINT AT 2,1;"TRIPUL. 1 HOM
BRE = 10 B/ORO."
4210 PRINT TAB 9;"CUANTOS TRIPUL
ANTES?"
4220 INPUT R
4225 IF FT+R>100 THEN PRINT AT 4
,8;"NO HAY LUGAR A BORDO"
4226 IF FT+R>100 THEN GOTO VAL "
4220"
4230 IF (O-10*R)<0 THEN GOTO VAL
 "4075"
4240 LET FT=FT+R
4245 PRINT AT 14,1;"    ";AT 14,
1;FT
4250 LET O=O-10*R
4260 PRINT AT 3,9;R;" NUEVOS HOM
BRES    "
4270 GOTO VAL "4080"
```



```
4300 PRINT AT 2,1;"REPAR. 20 $/O
RO POR PUNTO"
4310 PRINT TAB 8;"CUNTAS PTS. DE
 REP."
4320 INPUT R
4325 IF (E+R)>100 THEN GOTO VAL
"4320"
4330 IF (O-20*R)<0 THEN GOTO VAL
 "4075"
4340 LET E=E+R
4345 PRINT AT 16,1;"    ";AT 16,
1;100-E
4350 L"T O=O-20*R
4360 PRINT AT 3,8;R;" PUNTOS REP
ARADOS"
4370 GOTO VAL "4080"
4410 PRINT AT 5+L,5+K;D$(L,K)
4420 LET X2=X1
4425 LET X=X1
4430 LET Y2=Y1
4435 LET Y=Y1
4440 PLOT X2+12,Y2+2
4450 LET Z=0
4510 FOR R=8 TO 16 STEP 2
4520 PRINT AT R,0;"▌    "
4530 NEXT R
4540 PRINT AT 8,1;T;AT 10,1-(O>9
999);O;AT 12,1;C;AT 14,1;FT;AT 1
6,1;100-E
4550 IF Z=1 THEN RETURN
4760 IF E<5 THEN GOTO VAL "5750"
4770 IF FT<10 AND D$(L,K)=" " TH
EN GOTO 5500
4780 IF FT<1 THEN GOTO VAL "5250
"
4790 IF C=0 THEN GOTO VAL "5000"
4820 GOTO VAL "100"
5005 GOSUB VAL "6000"
5010 PRINT AT 2,1;"MOTIN LA TRIP
ULACION FURIOSA"
5020 PRINT TAB 7;"TOMA CONTROL D
E SU NAVE"
5030 PRINT TAB 7;"Y UD. SERA EL
ALMUERSO";TAB 9;"DE LOS TIBURONE
S"
5040 GOTO VAL "42"
5255 GOSUB VAL "6000"
5260 PRINT AT 2,1;"ABANDONADO TRI
PULACION INSUFI-"
5270 PRINT TAB 11;"CIENTE,COMO R
.CRUSOE"
5280 PRINT TAB 11;"UD.FLOTA SOLI
TARIO"
5290 GOTO VAL "42"
5505 GOSUB VAL "6000"
5510 PRINT AT 2,1;"DERIVA TRIPUL
ACION INSUFICI-"
5520 PRINT TAB 8;"ENTE PARA NAVE
GAR,SE-"
5530 PRINT TAB 8;"DIENTO Y FURIO
SO ESPERA";TAB 13;"SU MUERTE"
5540 GOTO VAL "42"
5755 GOSUB VAL "6000"
5760 PRINT AT 2,0;" ABANDONO  DE
MASIADAS AVERIAS"
5770 PRINT "EL BARCO  EMPIESA A
 HACER AGUA"
5780 PRINT TAB 11;"* LA NAVE SE
UNDE *"
5790 GOTO VAL "42"
6010 FOR R=1 TO 4
6020 PRINT AT R,1;"
 "
6030 NEXT R
6040 LET X2=X
6050 LET Y2=Y
6060 RETURN
6110 FOR R=1 TO 15
6120 PRINT AT 5+R,6;D$(R)
6130 NEXT R
6140 RETURN
7260 LET D$(1)="
 "
7265 LET D$(2)="
"+CHR$ Q(4)+" "
7270 LET D$(3)="
"+CHR$ Q(3)+"  "
7275 LET D$(4)="
 "
7280 LET D$(5)="     ==   ==
 "
7285 LET D$(6)="  ==      ==
 "
7290 LET D$(7)="  ==     ==
 "
7295 LET D$(8)="  ==  "+CHR$ Q(
1)+" ==      "
7300 LET D$(9)="    ===  ==
 "
7305 LET D$(10)="     =====
 =    "
7310 LET D$(11)="==
   "+CHR$ Q(2)     "
7315 LET D$(12)=" ==         =
 "
7320 LET D$(13)="      +++  =
 +"
7325 LET D$(14)="      +++  ==
 +"
7330 LET D$(15)="         ===
 **"
7420 GOSUB VAL "6100"
7450 RETURN
7510 LET D$(1)="
 "
7515 LET D$(2)="
 "
7520 LET D$(3)=" "+CHR$ Q(6)+
"  "
7525 LET D$(4)="
 "
7530 LET D$(5)="
 "
7535 LET D$(6)="
 "
7540 LET D$(7)="        "+CH
R$ Q(7)+"  "
7545 LET D$(8)="
 "
7550 LET D$(9)="  "+CHR$ Q(5)+"
 "
7555 LET D$(10)="
 "
7560 LET D$(11)="
 "
7565 LET D$(12)="
  "
7570 LET D$(13)="+
 "+CHR$ Q(8)+"  "
7575 LET D$(14)="+
  "
7580 LET D$(15)="**
 "+CHR$ Q(9)+" "
7620 GOSUB VAL "6100"
7650 RETURN
7760 LET D$(1)="
 "
7765 LET D$(2)="
 "
7770 LET D$(3)="
 "
7775 LET D$(4)="
 "
7780 LET D$(5)="
 "
7785 LET D$(6)="
 "
7790 LET D$(7)="          +++
 "
7795 LET D$(8)="   "+CHR$ Q(12)
+"    **  "
7800 LET D$(9)="        +
** "
7805 LET D$(10)="        + "
+CHR$ Q(13)+"++     "
7810 LET D$(11)="    "+CHR$ Q(11
)+"    ****  "
7815 LET D$(12)="           **
* "
7820 LET D$(13)="
 "
7825 LET D$(14)="  "+CHR$ Q(1
0)+"    "
7830 LET D$(15)="
 "
7920 GOSUB VAL "6100"
7950 RETURN
8010 LET D$(1)="       **** *
 ** "
8015 LET D$(2)="     *****"+
CHR$ Q(17)+"* ***   *"
8020 LET D$(3)="         ****
 **   **"
8025 LET D$(4)="         ***
 ****"
8030 LET D$(5)="         **
 *******"
8035 LET D$(6)="        **
 **"
8040 LET D$(7)="
****     *"
8045 LET D$(8)="  "+CHR$ Q(1
5)+"    **** *   *"
8050 LET D$(9)="         *
 *  *  "
8055 LET D$(10)="         *
 "+CHR$ Q(16)+"   =  "
8060 LET D$(11)="          =
 =  "
8065 LET D$(12)=" "+CHR$ Q(14)+"
     == ==  =  "
8070 LET D$(13)="
==  == =  "
8075 LET D$(14)="
 ==  "
8080 LET D$(15)="
 "
8200 GOSUB VAL "6100"
8240 RETURN
8260 LET D$(1)="
 ** "
8265 LET D$(2)="
 ***"
8270 LET D$(3)="  "+CHR$ Q(18)
+"       ****"
8275 LET D$(4)="
 * "
8280 LET D$(5)="
 "
8285 L"T D$(6)="
 "
8290 LET D$(7)="
 "
8295 LET D$(8)="
 "
8300 LET D$(9)="        "
+CHR$ Q(19)+"   "
8305 LET D$(10)="
"+CHR$ Q(20)+"      "
8310 LET D$(11)="
 "
8315 LET D$(12)="
"+CHR$ Q(22)+" "
8320 LET D$(13)="
   "+CHR$ Q(21)+" "
8325 LET D$(14)="
   "+CHR$ Q(23)+"  "
8330 LET D$(15)="
 "
8420 GOSUB VAL "6100"
8450 RETURN
8510 LET D$(1)="           ==
 == "
8515 LET D$(2)="*          ==
 "
8520 LET D$(3)="*          ==
 ==  "
8525 LET D$(4)="          == =
=="+CHR$ Q(26)+"==     "
8530 LET D$(5)="          ==
 ==    ="
8535 LET D$(6)="          ==
 ==  ==="
8540 LET D$(7)=" "+CHR$ Q(24)+
"        === == ===="
8545 LET D$(8)="
 ==  == "
8547 LET D$(9)="          ==
 ===  "
8550 LET D$(10)="           =
== =="
8555 LET D$(11)="        "+CHR$
 Q(25)+"     ==   "
8560 LET D$(12)="
 ==  "
8565 LET D$(13)="          **
****  "
8570 LET D$(14)="        +++
 **  "
8575 LET D$(15)="        ****
*  **  "
8600 GOSUB VAL "6100"
8740 RETURN
8755 CLS
8760 PRINT " VIAGE PELIGROSO**PO
R R.ORQUERA "
8770 FOR R=1 TO 4
8780 PRINT "▌";TAB 31;"▌"
8790 NEXT R
8810 PRINT "
 "
8820 FOR R=1 TO 15
8830 PRINT "      ";TAB 31;"█"
8840 NEXT R
8850 PRINT "
      MAPA 1 "
8860 PRINT AT 6,0;"ESTADO";AT 7,
1;"DIAS";AT 8,0;"▌0    "
8870 PRINT AT 9,1;"ORO ";AT 10,0
;"▌1000▌";AT 11,0;"COMIDA";AT 12
,0;"▌2500▌"
8880 PRINT AT 13,1;"TRIP.";AT 14
,0;"▌100 ▌";AT 15,1;"DANOS";AT 1
6,0;"▌0   ▌"
8890 PRINT AT 17,0;"      ";AT 18
,0;"▌ N ▌";AT 19,0;" O+E ";AT 20
,0;"▌ S ▌";AT 21,0;"      "
8980 GOSUB VAL "(7000+250*N)"
8985 PRINT AT 21,30;CHR$ (N+156)
8990 IF T>0 THEN GOTO 330
8992 PLOT X+12,Y+2
8994 GOTO VAL "100"
9510 PRINT TAB 1;"VIAGE PELIGROS
O**POR R.ORQUERA"
9520 PRINT " INSTRUCCIONES -"
9530 PRINT "UD.ES EL CAPITAN DEL
 HMS BOUNTY AMARRADO EN LA ESQUI
NA INFERIOR IZQUIERDA DEL MAPA 1
."
9540 PRINT "SU OVJETIVO ES REGRE
SAR CON LA  MAYOR CANTIDAD DE OR
O POSIBLE A SU PUERTO DE ORIGEN
(REPRESEN-  TADO POR ■ EN MAPA 6
),DISPO-    NIENDO EL CURSO Y SU
P. DEL VE-  LAMEN CORRECTOS Y TR
ATANDO DE   EVITAR LAS DIFICULTA
DES."
9550 PRINT "PREPARESE A SER ATAC
ADO POR PI- RATAS Y SALVAGES."
9560 PRINT "CADA TRIPULANTE COME
 UNA RACION POR DIA PERO PODRA C
OMPRAR ALI- MENTOS,REPARAR DANOS
 Y TOMAR    TRIPULANTES CUANDO T
OQUE TIERRA  Y SI SU CAPITAL LO
PERMITE."
9570 PRINT "ENCONTRARA MAS ORO E
N LOS TESO- ROS MARCADOS ■ AL ■.
"
9575 PRINT "*BUEN VIAJE*********
**********"
9740 RETURN
9760 LET FT=100
9770 LET N=1
9780 DIM D$(15,25)
9800 LET E=100
9810 LET O=1000
9820 LET C=2500
9830 LET X1=0
9840 LET Y1=0
9850 LET X=1
9860 LET Y=1
9870 LET T=0
9880 LET L=15
9890 LET K=1
9900 DIM Q(26)
9910 FOR R=1 TO 26
9920 LET Q(R)=165+R
9930 NEXT R
9940 LET P$="MALA NAVEGACION"
9950 LET Q$="DANOS AL BARCO"
9960 LET R$=" PUNTOS AVERIA"
9970 LET S$=" RACIONES"
9980 LET Z=0
9990 RETURN
```



Los ordenadores no muerden

Lynda COCCIONE y Gayle WINTER
Editorial ANAYA MULTIMEDIA
171 páginas

Aquí se aplica bien el famoso dicho de "los libros no muerden", sólo que en una versión muy actual. Su título original en inglés, "Computers won't byte" resulta más divertido por su doble sentido. Para manejar una computadora hoy en día, no es necesario conocer su funcionamiento interno, por las mismas razones que no es necesario saber de mecánica para conducir un automóvil (si bien nunca está de más).
En este libro encontramos una sencilla introducción a los ordenadores domésticos; explicación de términos informáticos; cómo saber si uno realmente está necesitando una computadora; lista de posibles usos; guía para comprar; y respuesta a las principales preguntas que se hace el principiante.
Dirigido tanto a los poseedores de home's o de PC's, está escrito en un lenguaje claro y divertido para todos aquellos que tímidamente se quieren acercar a la informática.



Spectrum, iniciación al código máquina

Antonio BELLIDO
Editorial PARANINFO
112 páginas

El código máquina sigue siendo sánscrito básico para los iniciados en la computación personal. Pero afortunadamente para todos, y en un lenguaje claro como es el de Antonio Bellido, se encara este tema de forma muy entendible.
Si bien se aconseja vivamente la ayuda de un programa ensamblador, también se dan instrucciones para programar en assembler en forma manual.
Sin embargo, según nuestra opinión, un libro solo no alcanza para dominar este lenguaje, todos resultan insuficientes por sí solos, debiendo tratar de ampliar estos conocimientos con mucha experimentación y más libros!
En términos de programación, nada hay más allá del código de máquina, lo que implica hablar "cara a cara" con el microprocesador, dice Antonio Bellido.



La Sinclair, proyectos de hardware / software

Ing. Pedro E. COLLA
Editorial HASA
152 páginas

La serie de máquinas compatibles o basadas en la pionera ZX81 han llegado a muchos usuarios en nuestro país, pero muchos que han descubierto en ellas sus grandes posibilidades extra, se han encontrado con grandes dificultades para obtener mayor información técnica.
El libro consta de una introducción al Z80, estructura, funcionamiento y circuito de la máquina, manejo de la RAM; proyectos con EPROM; interfases serie, paralelo y A/D D/A; síntesis de voz, y varios, con sus listados respectivos en assembler. Realmente recomendable para todo buen experimentador que posea conocimientos básicos de electrónica y coraje para desarmar su Sinclair o TK.



Micro Computadoras

Judy TATCHELL y Bill BENNETT
Editorial PLESA
48 páginas

Resulta sorprendente cómo en tan pocas páginas puede ponerse tanta información, será, tal vez, por aquello de que "una imagen vale por mil palabras". Si bien está presentado como un libro dirigido a los más chicos, su lectura no resulta para nada chocante ni infantil en sus términos.
En él se trata a la computadora vista por dentro y por fuera, dando las primeras explicaciones para programar, cómo ampliar los conocimientos adquiridos, otros usos, guía del comprador, glosario, gráficos y sonidos.

SORTEO K 64

EL PROBLEMA DEL MES

La aparición de las computadoras nos plantean un desafío y hasta un temor. Dejaremos de pensar alguna vez?

K64 sale al paso de esta cuestión, proponiéndole a nuestros lectores un pequeño desafío todos los meses para "forzarnos" a pensar un poco. Esto lo haremos planteando problemas cortos de lógica que difícilmente justifique realizarlo con la ayuda de la máquina. Pero así y todo está permitido hacer un poco de trampa con ese "cerebro electrónico", ya que esta trampa aún nos hará pensar más!

Resuelva el problema, coloque la respuesta en el cupón debajo completando los demás datos, y participe en el sorteo del "PROBLEMA DEL MES".

PROBLEMA DEL MES:

Un cazador de osos sigue las huellas de uno de éstos. Camina 1000 m. hacia el Sur, luego 1000 m. hacia el Este y luego 1000 m. hacia el Norte, volviendo al punto de partida y hallando al oso. De qué color era el oso?. Por qué?

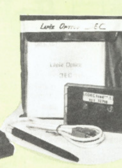
SOFTWARE PEEK



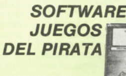
JOYSTICK PEEK

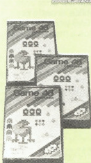


LAPIZ OPTICO DEC

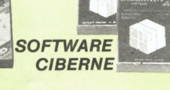


SOFTWARE JUEGOS DEL PIRATA





SOFTWARE CIBERNE





LIBROS

SOFTWARE GAME 48

DISTRIBUIDORA YENNY

PARTICIPE DEL SORTEO MENSUAL

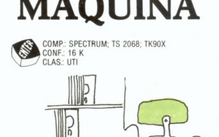
LLENE EL TALON Y ENVIARLO A: K64 (PROBLEMA DEL MES) PARANA 720 5° PISO BUENOS AIRES

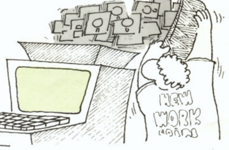
APELLIDO................................ NOMBRE.. FECHA..............................
DIRECCION.. LOCALIDAD..
PROVINCIA .... C.P. .... T.E. .... OCUPACION ....
EDAD .... AÑOS .... SEXO ....
COMPUTADORA:
CZ1000 ☐ CZ1500 ☐ CZ SPECTRUM ☐ TS2068 ☐ TK83 ☐ TK85 ☐ TK90 ☐ TK2000 ☐ C16 ☐ C64 ☐ C128 ☐ TI999 ☐ MSX ☐ TRS80 ☐
OTRAS .... MARCA .... NO POSEE ....
Piensa cambiar/comprar: .... MARCA ....
QUE TIPOS DE PROGRAMAS DESEA VER PUBLICADOS?:
JUEGOS ☐ EDUCATIVOS ☐ APLICACIONES COMERCIALES ☐ UTILITARIOS PROGRAMACION ☐
QUE ES LO QUE MAS LE GUSTA DE K64: ....
QUE ES LO QUE MENOS LE GUSTA: ....
OPINION GENERAL QUE LE MERECE K64: MUY BUENA ☐ BUENA ☐ REGULAR ☐ MALA ☐
USO DE SU COMPUTADORA: ENTRETENIMIENTO ☐ PROFESIONAL ☐ APRENDIZAJE ☐ HOBBY ☐ EXPERIMENTACION ☐ OTROS ☐

"LA SOLUCION DEL PROBLEMA ES: ....
....

CARGADOR DE CODIGO MAQUINA

COMP.: SPECTRUM; TS 2068; TK90X
CONF.: 16 K
CLAS.: UTI





Muy frecuentemente habrán visto programas que contienen rutinas en código máquina, pero ningún programa práctico para poder cargar en la memoria todos estos datos.

Para llevar a cabo esa empresa les ofrecemos este programa cargador de propósito general para aquellos que no posean mucha experiencia en lo que a lenguaje de máquina se refiera.

El programa nos interrogará primero acerca de la dirección de origen. Esta puede ser dada en decimal o hexadecimal, basta con indicarlo precediendo al valor con una H o una D, según corresponda.

Como muchos programas necesitan estar en un área segura de RAM, se ha incluido una opción para hacer un CLEAR en la dirección a la cual se comenzarán a escribir datos menos uno.

Algunas veces, los códigos están dados en hexadecimal y otras en decimal, por lo tanto el programa permite los dos tipos de ingreso. La principal diferencia es que al entrar códigos en decimal, sólo se puede hacer de a uno por vez, mientras que si se hace en hexadecimal, se pueden ingresar una cantidad de caracteres (sin espacios entre ellos) y el programa automáticamente tratará con mayúsculas y minúsculas.

La dirección y el código pokeado en la misma son puestos en pantalla para su inmediata verificación. En caso de cometer un error presione "s" para parar, tome nota de la dirección en la cual se cometió el error y recomience el programa con la dirección del error para corregirlo. No utilice en este caso la opción de CLEAR y continúe ingresando los datos a medida que sea necesario.

```
   1 REM ** K-64 **
   2 REM PROGRAMA PARA LA CARGA
DE CODIGO MAQUINA
   3 REM ************************
   4 REM PRESIONE S PARA PARAR
   5 REM ************************
  10 INPUT "Ingrese la direccion
 de origen  con H o D para indic
ar decimal ohexadecimal"; LINE a
$
  20 IF a$(1)="d" OR a$(1)="D" T
HEN LET addr=VAL a$(2 TO )
  30 IF a$(1)="h" OR a$(1)="H" T
HEN LET a$=a$(2 TO ): GOSUB 1000
: LET addr=a
  40 INPUT "Desea hacer un clear
 hasta esta direccion - 1(s/n)";
 LINE b$: IF b$="y" OR b$="Y" TH
EN CLEAR addr-1: LET addr=PEEK 2
3730+256*PEEK 23731+1
  50 INPUT "Desea entrar el codi
go en hexa  o en decimal (H o D)
"; LINE b$
 100 IF b$="d" OR b$="D" THEN GO
TO 200
 110 INPUT "Ingrese codigos hexa
"; LINE c$
 120 IF c$="s" OR c$="S" THEN ST
OP
 130 FOR j=1 TO LEN c$ STEP 2: L
ET a$=c$(j TO j+1): GOSUB 1000:
GOSUB 2000: NEXT J: GOTO 110
 200 INPUT "Ingrese codigos uno
a la vez"; LINE a$: IF a$="s" OR
 a$="S" THEN STOP
 210 LET a=VAL a$
 220 GOSUB 2000: GOTO 200
 999 STOP
1000 LET A=0: FOR I=LEN A$ TO 1
STEP -1: LET a=a+(CODE a$(i)-48-
(7 AND a$(i)>"9")-(32 AND a$(i)>
"#"))*(16↑((LEN a$)-i)): NEXT i
1010 RETURN
2000 POKE addr,a: PRINT addr;"-"
;a$: LET addr=addr+1: RETURN
```

C U R S O S



COMPUTACION

Enseñanza con

Commodore 64	Lenguajes
Texas TI 99	Cobol
Sinclair	Basic
Cobra 300	Logo

NIÑOS - JOVENES - ADULTOS

LAVALLEJA 100 - 854-7348 - 855-3562/0483



CURSOS:
- BASIC I
- BASIC II
- BASIC III
- USUARIOS

Los cursos se realizan con C-64, C-128 y monitor 40/80 columnas, un equipo por alumno. Atención individual

H & D electronics s.a.

Albarellos 1882 - (1640)

MARTINEZ - Tel. 792-1417

micros

computación en ZONA NORTE
BASIC - LOGO
TODAS LAS EDADES
CURSOS DE ESPECIALIZACION C-64
JUEGOS Y ACCESORIOS

Albarellos 2006 - Tel.: 792-0967
Martínez - Bs. As.

COMPUTACION

- *Cursos individuales y grupales*
- *Cursos para adolescentes y estudiantes*
- *Cursos para profesionales y empresas*
- *Prácticas en IBM PC, Commodore*
- *Trabajos Prácticos de la Facultad*

BASIC, FORTRAN, COBOL
Procesador de Palabra, Base de Datos, etc.

BECAS Y 1/2 BECAS

CPU - HUMAHUACA 4030 - CAPITAL
TE. 86-0716 - LU. a VI. de 17 a 21 HS.

CURSOS de COMPUTACION

- **PROGRAMACION BASIC** para Adultos
  Incluye:
  Manejo de Archivos y aplicaciones comerciales.
- **LOGO y BASIC** para niños y adolescentes, en la C-64 y 128. Movimientos de Figuras - Música.

I.D.E.S.I. *SANTA FE 1780 - P. 14 - Of. 1401/2*
Tel. 41-4507 de 11 a 20 hs.

¡ATENCION CHICOS! TE COPIAMOS UN JUEGO DE C-64 GRATIS A ELECCION, EN CADA CLASE

GENIAL

microsoft club
microcomputadoras
CURSOS

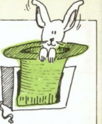
Gal. Cometa Loc. 17. Punta Alta, Prov. Bs. As. Inscripción 17 a 20 hs.

APRENDA COMPUTACION
EN UNA EMPRESA DE COMPUTACION
CON GENTE DE COMPUTACION

- CURSOS TEORICOS-PRACTICOS
- GRUPOS REDUCIDOS
- EQUIPOS DISPONIBLES PARA PRACTICAS
- POSIBILIDAD DE BECAS RENTADAS

INFORMES E INSCRIPCION:
PTE. R.S. PEÑA 950. CAPITAL TEL.: 35-6582/6465
PROMUEVEN: Q.B.S.A. Y SUPERMICRO S.A.

TRUCOS, TRAMPAS Y HALLAZGOS



Protegiendo programas

Como ya todos deben saber, si queremos que un programa se autoejecute una vez que se terminó de cargar, debemos grabarlo a sí mismo mediante una línea que contiene a la instrucción SAVE seguido del nombre del programa a salvar.
Ahora bien, si queremos cargar este programa y que no se autoejecute debemos hacer lo siguiente:
PONER LA MAQUINA EN FAST
PULSAR RAND USR 837
PONER EN MARCHA EL CASSETTE (PLAY)
Haciendo esto el programa se cargará pero no se ejecutará.
El comando RAND USR 837 es una llamada a la rutina de carga contenida en la rom. La utilidad de esta rutina se ve cuando queremos listar un programa que se autoejecuta y vemos que no hay forma de hacerlo.
Sin embargo, no todo está perdido para quienes desean mantener protegidos sus programas. Veamos el siguiente programa:

```
100 FAST
110 LET D=PEEK 16396
120 POKE 16396,0
130 SAVE "PRIV"
140 POKE 16396,D
150 SLOW
```

Este programa se graba a sí mismo en la línea 130, pero con el archivo de pantalla desarreglado, por la instrucción de la línea 120. Si se intenta cargarlo mediante el truco arriba mencionado, se producirá un CRACK. La autoejecución es necesaria para arreglar el archivo de la pantalla.

Análisis de sonido

Esta rutina en código máquina dibuja gráficos que representan el sonido que le llega a través del conector EAR. Para entrar el programa basta con ponerlo en una línea tipo 1 REM... y luego agregar:
2 RAND USR 16526
Antes de ejecutarlo con RUN debe ser grabado en cinta. La rutina tiene 45 caracteres y es como sigue:

```
01 00 FF DB FE 3C 20 01 C1 OF
F8 C9 16 00 DC 81 40 2A 25 40
2C C0 41 4A CB 38 CB 38 D5
CD B2 0B D1 14 D5 C7 72 C4
2A 0A D1 CB B2 18 E1
```

Como es lógico, al ejecutarlo sólo dibujará cuando el ZX 81 tenga presente algún sonido en conector EAR, que puede ser un programa, la radio o un cassette de música. Seguramente habrá que ajustar el volumen o el tono para que la gráfica tenga más definición. Cuando no haya sonido (o el volumen esté bajo) lo único que se verá es una línea en la parte inferior de la pantalla.
Puede hacerse que la gráfica sólo salga una vez y se interrumpa haciendo:
POKE 16551,192
POKE 16552,0
POKE 16553,0
Esta rutina puede servir de base para la utilización del ZX 81 como osciloscopio digital o, incluso, como frecuencímetro.

Enfasis

Este programa cambia el juego de caracteres del ZX SPECTRUM haciéndolos más gruesos y legibles. Para cambiar el juego de caracteres hay que hacer RAND USR 32700, una vez entrado y ejecutado el siguiente programa en BASIC:

```
10 CLEAR 32699
20 RESTORE
30 FOR N= 32700 TO 32747: READ A: POKE N,A: NEXT A
40 DATA
33,0,64,1,0,24,126,203,71,40,15,47,87,95,203,34,203,59,178,179,2
03,135,47,119,29,11,87,95,203,34,203,59,178,179,203,135,119,11,3
5,121,167,32,219,120,167,32,215,201
50 NEW
****************************************************************
+INT (B/H));CHR$ (28+B-H*INT (B/
H));
7050 IF B=118 THEN PRINT "*"
7055 IF B=203 OR B=221 OR B=237
OR B=253 THEN PRINT "%"
7060 NEXT M
7065 PRINT AT 21,0;"DIRECCION DE
 INICIO ";A;" FIN ";A+M
```

SUPLEMENTO

DE APLICACIONES

Comenzamos en esta página con un interesante proyecto para las CZ 2000, TK 90X o TS 2068. Luego, continuamos con la nota sobre la interfase paralelo para las TK 83 y CZ 1000, que nos permitirá múltiples usos (por ejemplo, hacer un secuenciador de luces programables). En software educativo, dos excelentes listados para TI 99, Spectrum y compatibles cuyos títulos hablan por sí solos: "Países, capitales y banderas" y "Huesos" (a no asustarse). Y a los "fans" de MSX les decimos cómo tener permanentemente un reloj en la pantalla cuando pasan largas horas frente al monitor.

CONVERSOR ANALOGICO DIGITAL

Presentamos una interfase con una rutina que nos será de gran utilidad en caso de tener que tomar datos externos, tales como temperatura, humedad o iluminación.

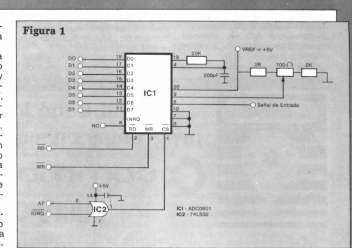
La necesidad de controlar dispositivos es casi tan antigua como la ciencia misma.

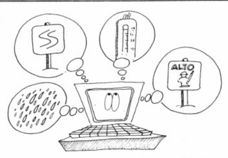
El control automático se basa esencialmente en un dispositivo "sensor" que toma información y se la entrega a un segundo elemento denominado "actuador", que es el que efectivamente produce resultados sobre cualquier situación o proceso bajo control.

El uso de computadores ha encontrado una masiva aceptación en ese tipo de aplicaciones, dado que se puede proveer de cierta "inteligencia" al sistema de control para tomar acciones en base a las situaciones que predetermínadamente se definan.

Los sistemas de control constituyen una rama de la tecnología lo suficientemente vasta, como para exceder ampliamente las posibi-

Figura 1





lidades de su visión, aunque sea superficial, en una publicación de esta índole.
Sin embargo, veremos en esta entrega y la siguiente dos elementos de capital importancia para lograr insertar un computador en este entorno.
Nos referimos a los conversores A/D (Analógico/Digital) y D/A (Digital/Analógico).
Si bien el computador es extremadamente útil adolece de un defecto de importancia, es digital, y el mundo que nos rodea salvando contadas excepciones no lo es; es en general analógico.
¿Cuál es la diferencia? Las señales digitales pueden tener solamente dos estados, ENCENDIDO y APAGADO ("1" o "0"); mientras que las analógicas pueden tener infinitos estados variando continuamente entre ellos.
Resulta obvio que para que un computador pueda interactuar eficazmente con el mundo exterior deberá poder "interpretar" las señales que de él provienen.
El conversor A/D es, esencialmente, la necesaria interfaz entre ambos mundos.
Su funcionamiento conceptual es sencillo, no todos los posibles estados que asume una variable analógica son igualmente importantes, normalmente carece de sentido práctico diferenciar entre algunos que están muy próximos entre sí.
Si imaginamos una señal formada de pequeños "escalones" lo suficientemente próximos entre sí, podremos dar una denominación a cada escalón.
Asignando una denominación numérica a cada uno no existe, en principio, dificultad para transformar la misma en un número binario el cual puede ser comprendido y evaluado por un computador.
La diferencia relativa (porcentual) entre el valor de un escalón y el siguiente se denomina RESOLUCION y determina que tan fiel es la representación numérica de la señal real, cuanto más próxima sea esta diferencia a cero más aproximada será la representación de la señal real.
Cuanto mayor sea el número de escalones con el que se pueda traducir una señal mayor será el número de señales distintas que se pueden procesar.
Supongamos que el computador utiliza 8 bits para representar estados, con este número de bits podrá definir 256 distintos, se dice en este caso que la resolución será del 0.3% (1/256=0.003).
Si el computador utilizara en cambio 16 bits para la conversión podría definir más de 65500 estados distintos por lo que la resolución sería mucho mejor.
La resolución depende entonces de la cantidad de bits involucrados, mayor la cantidad mejor la representación de la señal; el fin de la carrera es el costo; a mayor la cantidad de bits es también mayor el costo.
Para ámbitos como los que pueden ocupar la utilización de un computador hogareño la resolución dada por 8 bits será considerada como suficiente.
Existe un segundo factor a ser tenido en cuenta y es la velocidad de la conversión.
En efecto, el proceso de encontrar el número que mejor representa un estado analógico toma su tiempo; si la señal varía muy rápidamente no podrá físicamente representarla en forma correcta.
Cuanto menor es la frecuencia o velocidad de la señal menor costo tendrá el conversor para procesarla, nuevamente se entiende que a los efectos de las aplicaciones típicas de un computador hogareño, será suficiente la medición de valores contínuos o variando con frecuencias en hasta la gama del audio.
Un tercer factor a ser considerado es que no todas las señales susceptibles de ser convertidas tendrán una manifestación eléctrica; de hecho la mayoría de las posibiblemente interesantes no la tienen (temperatura, humedad, iluminación, etcétera).

Para salvar este inconveniente se utilizan dispositivos electroquímicos o electromecánicos denominados "transductores", los cuales tienen la habilidad de proporcionar una representación eléctrica de alguna otra variable que no lo es; por ejemplo, un micrófono permite traducir sonido (ondas de presión) en señales eléctricas que la representan.

La necesidad casi continua de realizar este tipo de transformaciones ha derivado en que los conversores A/D fueran migrando desde un conjunto de elementos discretos hasta un único circuito integrado; y posteriormente que el mismo bajara su costo a niveles razonables para la utilización cotidiana.

En la Fig. 1 se representa una interfaz de conversión A/D apta para ser utilizada por computadores TS2068 o Spectrum, aunque es posible su adaptación a muchos otros mediante conexiones apropiadas a las líneas de bus y la reescritura de la rutina de utilización.

D7) una representación digital del valor de la señal analógica.

Estando este dispositivo ideado para ser utilizado en el particular entorno del bus de un microprocesador, estas salidas son de tipo TRI-STATE por lo que solamente se conectan físicamente sobre el bus en determinados momentos que el procesamiento así lo requiere.

Para que ello ocurra el procesador debe indicarlo colocando las líneas CS y RD en bajo.

Entre el momento que se inicia la conversión y que la misma finaliza pasa un lapso de tiempo dado por la velocidad del conversor, dado que todo el proceso entre ambos puntos en el tiempo no están bajo el control del computador debe existir alguna forma mediante la cual el mismo se "entere" que ya existe alguna forma mediante la cual el mismo se "entere" que ya existe un valor convertido que debe ser leído.

Para ello se utiliza la línea denominada INTR (Interrupt Request), la misma está ideada para que se

se le debe afectar por un factor de escala para obtener la dimensión final de la señal analógica.

La misma dependerá del valor de la "referencia" del conversor que en este caso será de +5V.

Esto implica que sólamente se podrán convertir señales entre +0 y +5V con esta interfaz, el circuito acepta, sin inconvenientes referencias de hasta +12V, por su condición de marco de medición la tensión de referencia deberá ser una tensión de continua pura y estabilizada.

Para obtener el valor de la conversión se deberá realizar la siguiente operación con lo que devuelva la rutina. Tensión (V) =

$$\frac{[\text{Result. de conversión}] \times [\text{Ref.}]}{256}$$

Ejemplo: si el resultado de la conversión es 128, esto significa que, la señal a la entrada tenía +5 X 128 / 256 = 2.5 V

Para utilizar este enfoque debe tenerse en cuenta que la señal no podrá variar más rápidamente que lo que requiere el BASIC para realizar la anterior operación.

Software para interfaz A/D

```
00000 ;-----------------------------
00010 ; RUTINA DE UTILIZACION
00020 ; DE INTERFAZ A/D
00030 ; DEVUELVE UN VALOR ENTRE
00040 ; 0 Y 255 AL PROGRAMA
00050 ; BASIC A TRAVES DEL ACC BC
00060 ;-----------------------------
00070          ORG 30000
00080          ENT
00090 BEGIN    XOR A
00100 ;-----------------------------
00110 ; DISPARA LA CONVERSION
00120 ;-----------------------------
00125          DI
00130          OUT (#3F),A
00140 ;-----------------------------
00150 ; DEFINE LARGO DEL RETARDO
00160 ;-----------------------------
00170          LD BC,10000
00180 DELAY    DEC BC
00190          LD A,B
00200          OR C
00210          JR NZ,DELAY
00220 ;-----------------------------
00230 ; LEE AL FINALIZAR RETARDO
00240 ;-----------------------------
00250 READ     IN A,(#3F)
00260 ;-----------------------------
00270 ; COLOCA RESULTADO EN BC
00280 ;-----------------------------
00290          LD B,#00
00300          LD C,A
00310 ;-----------------------------
00320 ; RETORNA A BASIC
00330 ;-----------------------------
00340          EI
00350          RET
```

La misma se inserta en la arquitectura del computador como un port de I/O, aprovechando una de las direcciones que al efecto prevée el Z80; nótese también que el direccionamiento es el mismo que el utilizado para otros proyectos anteriores por lo que éste no podrá ser utilizado al mismo tiempo que ellos (interfaz de printer y de comunicaciones).

El integrado IC1 realiza todas las funciones necesarias de conversión. Para comenzar el procesador debe colocar simultáneamente las líneas WR y CS en bajo (Pines 3 y 1) con lo que se dispara el proceso de comparación; el algoritmo utilizado internamente por el dispositivo permite que con relativa velocidad se tenga disponibles en las líneas de salida (D0 a

la utilice en la generación de una interrupción al procesador; su inserción en un computador que utiliza esta facilidad para otros propósitos no es sencilla.

Sin embargo, existe otro método menos flexible pero más sencillo y adecuado en la mayoría de los casos, el mismo consiste en que luego de ordenar la iniciación de la conversión el procesador espere una cantidad fija de tiempo y luego de ello lea el resultado.

En la Fig. 2 se puede apreciar una corta rutina en lenguaje de máquina que sirve para controlar esta interfaz, las restricciones de velocidad impiden la utilización de BASIC para este propósito.

Esta rutina devolverá al BASIC un número entre 0 y 255 que será el resultado de la conversión; a este

Nótese por último que el programa se arma por encima de la dirección 32768 y que las interrupciones se desconectan mientras el procesador "espera" el resultado, esto tiene por objetivo el mejoramiento del "timming" de la conversión que de otra manera se vería alterado por conflictos con la generación de video.

La construcción podrá encararse mediante técnicas de wire-wrapping o soldado punto a punto; y la conexión física sobre el computador se hará a través de un conector apropiado al peine de expansión.

Debe tenerse precaución en no exceder con la señal a convertir el rango mencionado previamente por las consecuencias que ello acarrearía sobre el computador.

INTERFASE CONTROL PARA LA CZ 1000/1500; TK 83/85

II parte – Conclusión

Programando el PIO:



Dado que la versión del basic del ZX 81 no ofrece la posibilidad de comunicarse con dispositivos externos en forma directa, debemos recurrir a una rutina en código máquina para programar y operar el PIO. Esta deberá ser cargada en la memoria de la máquina cada vez que se utilice la interfase, dado que sin sus instrucciones el PIO no podría operar. Las palabras de dato y control pueder ser pasadas desde basic mediante pokes apropiados. Esta rutina es ejecutada mediante la instrucción USR. Si bien no es imprescindible, algún conocimiento del código máquina del Z 80 nos ayudará a entender el funcionamiento de la rutina.

Los datos son transferidos de la CPU al PIO direccionando uno de sus registros y luego escribiendo en él mediante la instrucción OUT del Z 80. Tan sólo necesitaremos encargarnos de cuatro registros del PIO para esta aplicación; éstos son: port A control, por B control, port A data, y por B data. Cada registro es accedido por una única dirección. Las instrucciones de entrada y salida están siempre asociadas a direcciones de un byte. Este será el que forma los ocho bits más bajos del bus de address.

Para operar el PIO se necesitan un mínimo de tres bits. Normalmente la línea de address A0 se conecta a la línea de selección de port (B/A SEL) del PIO y la línea A1 es conectada a la línea de selección de control de datos (C/D SEL). Los seis restantes bits del byte de direccionamiento son utilizados para seleccionar uno de varios dispositivos de entrada-salida. Dado que el PIO es el único dispositivo de entrada-salida que disponemos en este sistema, la decodificación no es necesaria. Como se puede ver en la figura 2, la línea de address A7 es invertida y conectada directamente al chip enable (CE) del PIO.

Por lo tanto, cualquier dirección en el rango de 10000000 a 11111111 habilitará al PIO. La rutina en lenguaje de máquina utiliza la instrucción:

OUT(n),A

Listado de la rutina en código máquina

Localización	Código Dec	Código Hex	Mnemónico
0	0	00	NOP
1	0	00	NOP
2	62	3E	LD A,CFH
3	207	CF	
4	211	D3	OUT (82H),A
5	130	82	
6	62	3E	LD A,00H
7	0	00	
8	211	D3	OUT (82H), A
9	130	82	
10	62	3E	LD A,CFH
11	207	CF	
12	211	D3	OUT (83H),A
13	131	83	
14	63	3E	LD A,FFH
15	255	FF	
16	211	D3	OUT (83H),A
17	131	83	
18	62	3E	LD A,07H
19	07	07	
20	211	D3	OUT (82H),A
21	130	82	
22	211	D3	OUT (83H),A
23	131	83	
24	201	C9	RETN
25	62	3E	LD A,00H
26	00	00	
27	211	D3	OUT (80H),A
28	128	80	
29	201	C9	RETN
30	33	21	LD HL,0000H
31	00	00	
32	0	00	
33	14	0E	LD C,81H
34	129	81	
35	237	ED	IN L,(C)
36	104	68	
37	201	C9	RETN

El número posible de aplicaciones está limitado sólo por la imaginación. Permite conectar a la computadora con conversores analógicos digitales, o hacer un secuenciador de luces programables.

Esta transfiere el contenido del acumulador (registro A interno del CPU) al dispositivo de entrada-salida representado por n. En la tabla 1 podemos ver las direcciones de los registros internos del PIO y su significado cuando se utilizan con la instrucción OUT.

Antes de que los datos puedan ser mandados al port, algunas palabras de control deben ser cargadas en los registros internos del PIO. Este proceso se denomina inicialización y el programa que lo lleva a cabo es llamada la rutina de inicialización. Varias cosas deben ser llevadas a cabo en este proceso de inicialización; debe setearse el modo de operación, la dirección de datos debe ser establecida, y las interrupciones deben ser controladas. En este ejemplo, la selección del modo tres simplifica esto dado que las líneas de handshake no son utilizadas. El modo de operación es seleccionado escribiendo una palabra de control con los cuatro bits menos significativos altos. Los dos bits más significativos determinan el modo de operación y los demás bits no son utilizados, como se puede ver en la figura 5. Cuando el modo de control (modo 3) es seleccionado para un port en particular, la próxima palabra de control debe definir la dirección de los datos en las líneas de dicho port. Cada línea corresponde a un bit en la palabra de control, el bit más significativo de la palabra de control corresponde a la línea más significativa de I/O. Una condición alta (1) significará salida, mientras que una baja (0) será una entrada. Por ejemplo, supongamos que la palabra de control F0H es utilizada para seleccionar la dirección de datos en el port B. Las líneas PB0 a PB3 serán dispuestas para salida, mientras que las PB4 a PB7 lo serán para entrada.

Figura 4

Dirección Binario	Hexa	Decimal	Contenido del acumulador interpretado como:
1XXXXX00	80	128	DATO-PORT A
1XXXXX01	81	129	DATO PORT B
1XXXXX10	82	130	CONTROL -PORT A
1XXXXX11	83	131	CONTROL -PORT B
0XXXXXXX	00	0	PIO DESHABILITADO ,SIN CAMBIO

Figura 5

Modo de operación	Binario	Palabra de control Hexa	Decimal
SALIDA	0 00XX1111	0F	15
ENTRADA	1 01XX1111	4F	79
BIDIRECCIONAL	2 10XX1111	8F	143
CONTROL	3 11XX1111	CF	207

Las interrupciones son manejadas en una forma muy simple en esta aplicación; son deshabilitadas escribiendo 07H a los registros de control en ambos ports.

La rutina para el manejo del PIO está listada en la figura 4 y puede ser ubicada en cualquier espacio vacío de la memoria. El hecho de que no debe ser ubicada en un lugar en especial significa que es relocateable, lo que nos permite una mayor flexibilidad de trabajo. De todos modos, para que esta sea salvada en cassette un buen lugar para ubicarla puede ser en una línea de tipo 1 REM.... Una rutina para cargar un código de esta forma no es complicada y ya fue publicada en números anteriores. Para referirnos al inicio de la rutina

utilizaremos el símbolo ORG. Este debe ser suministrado por el usuario y en caso de almacenar la rutina en una línea 1 REM su valor será 16514. Debemos tener cuidado con este valor dado que todas las direcciones en la rutina deberán ser referidas a ORG.
El programa consiste en tres rutinas separadas, cada una de ellas con un return al final de la misma que hace volver al programa en basic. La primera de estas rutinas inicializa el PIO, seteando el port A para salida y el port B para entrada. Las interrupciones son deshabilitadas en la última parte de la rutina de inicialización. La siguiente rutina envía un dato seleccionado al port A. Esta es modificada por medio de un poke desde el BASIC. La tercer rutina, la rutina de lectura del port B, lee los datos presentes en las líneas del port B y guarda este dato en el registro 1. El par HL es reseteado al comienzo de la rutina. El guardar la información en el registro 1 es conveniente, dado que al efectuar una instrucción USR desde BASIC, ésta retorna con el valor del registro HL que en nuestro caso coincidirá con el dato leído en el port B.
El programa en BASIC que ejecuta la rutina en máquina nos permitirá tanto escribir como leer datos del PIO. Esto se lleva a cabo en forma muy sencilla. En el caso de la escritura de un valor al PIO, se hace un poke a la dirección en que se encuentra el dato a enviar en la instrucción OUT y luego se llama a las rutinas de inicialización y salida. Para el caso de leer un dato, se utiliza directamente una instrucción de la forma LET X= USR... por el motivo explicado anteriormente. En la figura 7 tenemos un cuadro que nos resume las variables del programa para una mejor comprensión del mismo.
Cuando deseamos enviar un dato como salida, debemos introducir un valor entre 0 y 255. Por ejemplo, si ingresamos el valor 255, todas las líneas del port A serán puestas en 1. Seleccionando la opción de lectura tendremos en pantalla éstas altas y las demás conectadas a masa, deberemos leer el valor 129.

Figura 6

```
2 PRINT "ORG...?"
3 INPUT ORG
4 LET AO=ORG+25
5 LET BI=ORG+30
6 LET MLA=AO+1
7 CLS
10 PRINT "MENU"
20 PRINT "1) PORT -A- SALIDA"
30 PRINT "2) PORT -B- ENTRADA"
40 INPUT A
50 LET A=A*1000
60 CLS
70 GOTO A
1000 PRINT "INGRESE BYTE DE SALIDA"
1010 INPUT B
1020 POKE MLA,B
1030 LET X=USR(ORG)
1040 LET X=USR(AO)
1050 PRINT "ORTO BYTE ?"
1060 INPUT Z$
1070 CLS
1080 IF NOY Z$="" THEN GOTO 10
1090 GOTO 10000
2000 PRINT "APRETE ENTER PARA LEER EL PORT
     B"
2010 INPUT Z$
2020 CLS
2030 IF NOT Z$="" THEN GOTO 10
2040 LET X= USR (ORG)
2050 LET X=USR(BI)
2060 PRINT "DATO EN EL PORT B...";X
2070 GOTO 2000
```

Figura 7

Variable	Significado
ORG	PRINCIPIO DE LA RUTINA DE INICIALIZACION
AO	PRINCIPIO DE LA RUTINA DE SALIDA DEL PORT A
BI	PRINCIPIO DE LA RUTINA DE ENTRADA DEL PORT B
MLA	DIRECCION DEL BYTE DE SALIDA POR PORT A

Aplicaciones:

El número de posibles aplicaciones está solamente limitado a la imaginación. Con 16 líneas de entrada/salida no existe inconveniente alguno para interfasear a la computadora con conversores A/D (analógico digital) o D/A (digital analógico), o controlar motores y aparatos de mayor potencia mediante el uso de un amplificador y un relay. Para aquellos que se interesen en la electrónica, esta interfase nos brinda innumerables posibilidades, como por ejemplo, tener un sofisticado y flexible generador de funciones programable, con la salvedad de que no sólo se pueden generar las formas de onda habituales, sino también cualquier otra que se nos ocurra. Podemos también hacer un secuenciador de luces programable y muchos usos más.
Esperamos que con este proyecto tendrán diversión para rato y les rogamos que revisen bien todo el cableado y conexiones antes de conectar la computadora y que tengan cuidado si se deciden a trabajar controlando la tensión de línea.

GUIA PRACTICA

PAISES BANDERAS Y CAPITALES

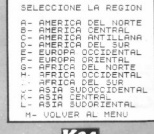
COMP.: SPECTRUM; TS 2068; TK 90X
CONF.: 48 K
CLAS.: EDU
AUTOR: Guillermo Jamilis
Lomas del Mirador - Pcia. Bs. As.



La cambiante situación política del mundo nos obliga a actualizar nuestro conocimiento de la geografía política, ya que en forma contínua los movimientos emancipadores de los pueblos tienden a independizar a los estados, variando así el número de naciones que pueblan el mundo.
Este juego nos ilustra acerca de la composición actual de los países que integran el planeta, de cuáles son sus ciudades capitales, y del símbolo que las distingue: la policromía de sus banderas.
El conocimiento de las naciones que pueblan la Tierra permitirá un mayor acercamiento cultural y económico, promoviendo de tal modo un mayor progreso material y espiritual de las naciones.
El programa provee un listado completo de los países que integran el mundo actual, así como las principales territoriales de países extracontinentales.

Por cada país o posesión se indica la ciudad capital. También se muestran las banderas de la mayoría de los diferentes países, a fin de aprender a individualizarlas.
Se complementa con un juego o competencia, en el cual pueden intervenir amigos y familiares, haciendo del aprendizaje un agradable entretenimiento.
Se inicia con un menú principal, donde se ofrecen las siguientes opciones:
1.- INSTRUCCIONES: a través de ellas, los jugadores serán guiados para que puedan obtener la imagen de las diferentes pantallas.
2.- LISTADO DE PAISES Y CAPITALES: Eligiendo esta opción, aparecerán todos los países del mundo, agrupados por regiones, y sus respectivas ciudades capitales.
3.- MOSTRAR BANDERAS: Esta opción, permite visualizar en un T.V. color, la policromía de las banderas que individualizan a los diferentes estados. Las banderas se cambian oprimiendo una tecla.
4.- COMPETENCIA: La opción, permite a los participantes introducirse en un entretenido y apasionante juego que, además de educativo, ampliará la perspectiva individual sobre los problemas mundiales.
Pueden participar en el juego cualquier cantidad de personas, pero en la tabla de colocaciones, aparecerán sólo los diez participantes que hubieran obtenido mayor puntaje.
El juego se compone de dos etapas: En la primera, se informa sobre un país. La máquina nos pregunta cuál es su capital y en qué continente queda esa nación. Cada respuesta correcta suma diez puntos al jugador. Cada respuesta incorrecta agrega un punto a las pérdidas.
Después de cada respuesta, se debe oprimir la tecla ENTER. También debe oprimirse la tecla ENTER o Y cuando la computadora pregunta: "scroll?".
Quienes no pierdan cinco veces en la primera etapa, pasan a la segunda, que consiste en la aparición al azar de distintas banderas. Aquí deberemos indicar a qué país corresponde, cuál es la capital de dicho país y en qué continente se encuentra.
Si compitiendo se acumulan cinco errores, el juego se termina. Deberá jugarse de a un participante por vez.
Para grabar el programa después de haberlo tipeado en la computadora, lo haremos con SAVE "PAISES" LINE 1. Esto permitirá que el programa comience al terminar la carga (LOAD"").
ACLARACION: Después de haber pasado el programa por la impresora se agregaron algunas instrucciones VAL a los efectos de ahorrar memoria que no alteran el funcionamiento del programa.





```
1 REM ++++++++++++++++++++++++++
2 REM + PAISES Y CAPITALES +
3 REM + © GUILLERMO JAMILIS +
4 REM ++++++++++1985++++++++++
7 FOR I=0 TO 7: READ A: POKE USR "N"+I,A: NEXT I
8 DATA 60,66,98,82,74,70,66,0
9 DIM R(VAL "10"): DIM R$(VAL "10",VAL "10"): POKE VAL "23658",VAL "8": DIM P$(VAL "15"): DIM C$(VAL "16"): DIM X$(VAL "5",VAL "8"): RESTORE VAL "5010": FOR I=VAL "1" TO VAL "5": READ X$(I): NEXT I
10 PAPER VAL "7": BORDER VAL "5": INK VAL "0": CLS
20 PRINT AT VAL "1",VAL "8";"PAISES DEL MUNDO";AT VAL "3",VAL "6";"CAPITALES Y BANDERAS"
30 INK VAL "2": PRINT AT VAL "5",VAL "3"; INK VAL "3";"SELECCIONE UNA OPCION": PRINT AT VAL "9",VAL "3";"1- INSTRUCCIONES";AT VAL "11",VAL "3";"2- LISTADO DE PAISES";AT VAL "12",VAL "9";"Y CAPITALES";AT VAL "14",VAL "3";"3- MOSTRAR BANDERAS";AT VAL "16",VAL "3";"4- COMPETENCIA"
40 LET K=CODE INKEY$: IF K<VAL "49" OR K>VAL "52" THEN GO TO VAL "40"
50 RESTORE VAL "4952"+K: READ K: GO TO K
60 RESTORE VAL "5100": PAPER VAL "6": BORDER VAL "4": INK VAL "1": CLS
70 PRINT AT VAL "1",VAL "7";"PAISES Y CAPITALES"; INK VAL "2"''TAB VAL "4";"SELECCIONE LA REGION"'': FOR I=VAL "1" TO VAL "13": BEEP VAL ".01",I: INK VAL "0": PRINT TAB VAL "4";CHR$ (VAL "64"+I);"- ";: READ Z$,NP,C: PRINT Z$: NEXT I: PRINT AT VAL "20",VAL "4";"N- VOLVER AL MENU"
80 LET K=CODE INKEY$: IF K<VAL "65" OR K>VAL "78" THEN GO TO VAL "80"
85 IF K=VAL "78" THEN GO TO VAL "10"
90 RESTORE VAL "5035"+K: READ Z$,NP,C
100 CLS : PRINT INK VAL "1";Z$''"PAIS","CAPITAL"''
110 FOR I=VAL "1" TO NP: RESTORE C+I-VAL "1": READ P$,C$: PRINT P$,C$: NEXT I
120 BEEP VAL ".02",VAL "14": PRINT '" PULSE UNA TECLA PARA CONTINUAR": PAUSE 0: GO TO VAL "60"
130 RESTORE VAL "5100": PAPER VAL "7": BORDER VAL "6": INK VAL "0": CLS
140 PRINT AT VAL "1",VAL "7";"PAISES Y CAPITALES": INK VAL "2"''TAB VAL "4";"SELECCIONE LA REGION"'': FOR I=VAL "1" TO VAL "12": BEEP VAL ".01",I: INK VAL "0": PRINT TAB VAL "4";CHR$ (VAL "64"+I);"- ";: READ Z$,NP,C: PRINT Z$: NEXT I: PRINT AT VAL "20",VAL "4";"M- VOLVER AL MENU"
150 LET K=CODE INKEY$: IF K<VAL "65" OR K>VAL "77" THEN GO TO VAL "150"
160 IF K=VAL "77" THEN GO TO VAL "10"
170 RESTORE VAL "5035"+K: READ P$,NP,COM
180 FOR Z=VAL "1" TO NP
190 RESTORE COM+Z-VAL "1": READ P$,C$,C,B: IF B=VAL "9" THEN GO TO VAL "210"
200 PAPER VAL "7": BORDER B: CLS : GO SUB VAL "3000": BEEP VAL ".02",VAL "14": PAPER VAL "7": INK VAL "0": PRINT AT VAL "21",VAL "0";P$,"PULSE UNA TECLA": PAUSE VAL "0"
210 NEXT Z: GO TO VAL "130"
220 LET CNT=VAL "10": PAPER VAL "1": BORDER VAL "1": INK VAL "7": CLS
230 PRINT AT VAL "1",VAL "7";"PAISES   CAPITALES": FOR I=VAL "1" TO VAL "10": PRINT AT I+VAL "3",VAL "7";R$(I);"...";TAB VAL "25"-LEN STR$ R(I);R(I): NEXT I
240 PRINT AT VAL "15",VAL "10"; FLASH VAL "1";"COMPETENCIA"; FLASH VAL "0";AT VAL "18",VAL "4";"PULSE ""C"" PARA COMENZAR"
250 IF INKEY$<>"C" THEN GO TO VAL "250"
260 CLS : LET P=VAL "0": LET PD=VAL "0"
270 PRINT "PUNTOS: ";TAB VAL "14"-LEN STR$ P;P;TAB VAL "20";"PERDIDAS: ";PD;AT VAL "5",VAL "2";"PAIS: ";AT VAL "8",VAL "2";"CAPITAL: ";AT VAL "10",VAL "2";"CONTINENTE: ";AT VAL "21",VAL "0";"PAISES: "
280 FOR V=VAL "1" TO CNT: RESTORE VAL "6000"+INT (RND*VAL "189"): READ P$,C$,C
290 PRINT AT VAL "5",VAL "8";P$;AT VAL "8",VAL "11";TAB VAL "27";AT VAL "10",VAL "14";TAB VAL "22";AT VAL "21",VAL "8";V
300 DIM K$(VAL "16"): INPUT "CAPITAL? ";K$: PRINT AT VAL "8",VAL "11";K$: IF K$=C$ THEN PRINT AT VAL "12",VAL "12"; FLASH VAL "1";"CORRECTO": LET P=P+VAL "10": GO TO VAL "320"
310 LET PD=PD+VAL "1": PRINT AT VAL "12",VAL "11";"INCORRECTO";AT VAL "14",VAL "3";"LA RESPUESTA CORRECTA ES";AT VAL "15",VAL "8";C$
320 PAUSE VAL "300": PRINT AT VAL "8",VAL "11";C$;AT VAL "12",VAL "11";TAB VAL "22";AT VAL "14",VAL "3";TAB VAL "31";AT VAL "15",VAL "8";TAB VAL "24"
330 PRINT AT VAL "0",VAL "14"-LEN STR$ P;P;AT VAL "0",VAL "30";PD: IF PD<VAL "5" THEN GO TO VAL "390"
340 PRINT AT VAL "12",VAL "7"; BRIGHT VAL "1";"TERMINO EL PARTIDO"
350 IF P<=R(VAL "10") THEN PAUSE VAL "300": GO TO VAL "10"
360 LET R(VAL "10")=P: FOR I=VAL "10" TO VAL "2" STEP VAL "-1": IF R(I)>R(I-VAL "1") THEN LET N=R(I-VAL "1"): LET N$=R$(I-VAL "1"): LET R(I-VAL "1")=R(I): LET R$(I-VAL "1")=R$(I): LET R$(I)=N$: LET R(I)=N: NEXT I
370 PRINT AT VAL "14",VAL "9";"UD. HA LOGRADO";AT VAL "15",VAL "8";"LA ";I;"A. UBICACION";AT VAL "16",VAL "9";"EN LA TABLA DE";AT VAL "17",VAL "8";"PUNTAJES MAXIMOS";AT VAL "18",VAL "13";"DE HOY"
380 INPUT "INGRESE SU NOMBRE"'"(MAX.10 LETRAS) ";R$(I): GO TO VAL "10"
390 DIM K$(VAL "8"): INPUT "CONTINENTE? ";K$: PRINT AT VAL "10",VAL "14";K$: FOR I=VAL "1" TO VAL "5": IF K$<>X$(C) THEN NEXT I: PRINT AT VAL "12",VAL "11";"INCORRECTO";AT VAL "14",VAL "3";"LA RESPUESTA CORRECTA ES";AT VAL "15",VAL "8";X$(C): LET PD=PD+VAL "1": GO TO VAL "400"
395 PRINT AT VAL "12",VAL "12"; FLASH VAL "1";"CORRECTO": LET P=P+VAL "10"
400 PAUSE VAL "300": PRINT AT VAL "12",VAL "11";TAB VAL "25";AT VAL "14",VAL "3";TAB VAL "31";AT VAL "15",VAL "8";TAB VAL "24"
410 PRINT AT VAL "0",VAL "14"-LEN STR$ P;P;AT VAL "0",VAL "30";PD: IF PD<VAL "5" THEN NEXT V: GO TO VAL "430"
420 GO TO VAL "340"
430 PAPER VAL "4": BORDER VAL "4": INK VAL "0": CLS
440 PRINT AT VAL "2",VAL "6";"TERMINO LA 1A. ETAPA";AT VAL "5",VAL "6";"UD. LOGRO ";P;" PUNTOS";AT VAL "10",VAL "2"; INK VAL "1";"AHORA DEBE RECONOCER EL PAIS";AT VAL "12",VAL "5";"A PARTIR DE SU BANDERA";AT VAL "15",VAL "8";"SERAN ";CNT;" PAISES"
450 PAUSE VAL "500"
460 FOR V=VAL "1" TO CNT
470 RESTORE VAL "6000"+INT (RND*VAL "176"): READ P$,C$,CO,B: IF B=VAL "9" THEN GO TO VAL "470"
480 BORDER B: PAPER VAL "7": INK VAL "0": CLS : GO SUB VAL "3000": BEEP VAL ".02",VAL "14"
490 DIM K$(VAL "15"): INPUT "PAIS? ";K$
500 OVER VAL "0": PAPER VAL "7": INK VAL "0": CLS : PRINT "PUNTOS: ";TAB VAL "14"-LEN STR$ P;P;TAB VAL "20";"PERDIDAS: ";PD;AT VAL "3",VAL "2";"PAIS: ";K$;AT VAL "6",VAL "2";"CAPITAL: ";AT VAL "8",VAL "2";"CONTINENTE: ";AT VAL "21",VAL "0";"PAISES: ";: IF K$=P$ THEN PRINT AT VAL "11",VAL "12"; FLASH VAL "1";"CORRECTO": LET P=P+VAL "20": GO TO VAL "520"
510 PRINT AT VAL "11",VAL "11";"INCORRECTO";AT VAL "13",VAL "3";"LA RESPUESTA CORRECTA ES";AT VAL "14",VAL "8";P$: LET PD=PD+VAL "1"
520 PAUSE VAL "300": PRINT AT VAL "3",VAL "8";P$;AT VAL "0",VAL "14"-LEN STR$ P;P;AT VAL "0",VAL "30";PD;AT VAL "11",VAL "11";TAB VAL "22";AT VAL "13",VAL "3";TAB VAL "30";AT VAL "14",VAL "8";TAB VAL "24": IF PD=VAL "5" THEN GO TO VAL "340"
530 DIM K$(VAL "16"): INPUT "CAPITAL? ";K$: PRINT AT VAL "6",VAL "11";K$: IF K$=C$ THEN PRINT AT VAL "11",VAL "12"; FLASH VAL "1";"CORRECTO": LET P=P+VAL "10": GO TO VAL "550"
540 PRINT AT VAL "11",VAL "11";"INCORRECTO";AT VAL "13",VAL "3";"LA RESPUESTA CORRECTA ES";AT VAL "14",VAL "8";C$: LET PD=PD+VAL "1"
550 PAUSE VAL "300": PRINT AT VAL "6",VAL "11";C$;AT VAL "0",VAL "14"-LEN STR$ P;P;AT VAL "0",VAL "30";PD;AT VAL "11",VAL "11";
```



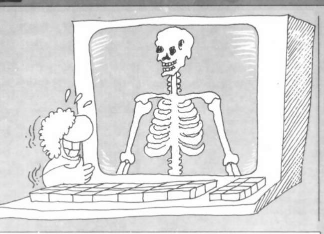

```
TAB VAL "22";AT VAL "13",VAL "3";TAB VAL "30";AT VAL "14",VAL "8";TAB VAL "24": IF PD=VAL "5" THEN GO TO VAL "340"
 560 DIM K$(VAL "8"): INPUT "CONTINENTE? ";K$: PRINT AT VAL "8",VAL "14";K$: IF K$=X$(CO) THEN PRINT AT VAL "11",VAL "12"; FLASH VAL "1";"CORRECTO": LET P=P+VAL "10": GO TO VAL "580"
 570 PRINT AT VAL "11",VAL "11";"INCORRECTO";AT VAL "13",VAL "3";"LA RESPUESTA CORRECTA ES";AT VAL "14",VAL "12";X$(CO): LET PD=PD+VAL "1"
 580 PAUSE VAL "300": PRINT AT VAL "0",VAL "14"-LEN STR$ P;P;AT VAL "0",VAL "30";PD;AT VAL "11",VAL "11";TAB VAL "22";AT VAL "13",VAL "3";TAB VAL "30";AT VAL "14",VAL "8";TAB VAL "24": IF PD=VAL "5" THEN GO TO VAL "340"
 590 NEXT V
 600 LET CNT=CNT+VAL "10": IF CNT=VAL "210" THEN LET CNT=VAL "200"
 610 CLS : PRINT AT VAL "2",VAL "6"; INK VAL "1";"TERMINO LA 2A. ETAPA";AT VAL "5",VAL "6"; INK VAL "2";"UD LOGRO ";P;" PUNTOS";AT VAL "10",VAL "8";"SERAN ";CNT;" PAISES"
 620 PAUSE VAL "500": PAPER VAL "1": BORDER VAL "1": INK VAL "7": CLS : GO TO VAL "270"
 630 PAPER VAL "5": BORDER VAL "5": INK VAL "0": CLS
 640 PRINT AT VAL "1",VAL "9"; INK VAL "2";"INSTRUCCIONES"
 650 PRINT AT VAL "3",VAL "3";"OPCION 2 DEL MENU:";AT VAL "5",VAL "3"; FLASH VAL "1"; INK VAL "1";"LISTADO DE PAISES Y CAPITALES": PRINT AT VAL "7",VAL "5";"APARECERA UNA LISTA CON LAS   REGIONES DEL MUNDO";AT VAL "10",VAL "5";"OPRIMA LA LETRA CORRESPON-   DIENTE A LA OPCION DESEADA Y   APARECERA EN LA PANTALLA EL   LISTADO DE LOS PAISES DE ESA   REGION Y SUS CAPITALES ";AT VAL "21",VAL "8";"PULSE UNA TECLA": PAUSE VAL "0"
 660 CLS : PRINT AT VAL "3",VAL "3";"OPCION 3 DEL MENU:";AT VAL "5",VAL "3"; FLASH VAL "1"; INK VAL "1";"MOSTRAR BANDERAS": PRINT AT VAL "7",VAL "5";"APARECERAN LAS REGIONES DEL   MUNDO";AT VAL "10",VAL "5";"OPRIMA LA TECLA QUE CORRES-  PONDE A ESA OPCION Y VERA LAS  BANDERAS DE LOS PAISES DE ESA  REGION.";AT VAL "21",VAL "8";"PULSE UNA TECLA": PAUSE VAL "0"
 670 CLS : PRINT AT VAL "3",VAL "3";"OPCION 4 DEL MENU:";AT VAL "5",VAL "3"; FLASH VAL "1"; INK VAL "1";"COMPETENCIA": PRINT AT VAL "7",VAL "5";"ELIJA ESTA OPCION DESPUES     DE HABER ESTUDIADO LOS PAISES   SUS CAPITALES Y BANDERAS";AT VAL "11",VAL "5";"INICIE CON SUS FAMILIARES      Y AMIGOS LA COMPETENCIA SOBRE   LOS PAISES Y SUS CAPITALES."
 680 PRINT AT VAL "15",VAL "5";"LA COMPETENCIA FINALIZA AL    PERDER 5 VECES.";AT VAL "21",VAL "8";"PULSE UNA TECLA": PAUSE VAL "0": GO TO VAL "10"
3000 READ NP
3010 FOR H=1 TO NP: READ NSP,CP: DIM C(CP,12)
3020 FOR I=1 TO CP: READ C(I,1),C(I,2): LET B=3
3030 FOR J=1 TO C(I,2): READ C(I,B),C(I,B+1): LET B=B+2
3040 NEXT J: NEXT I
3050 FOR I=1 TO NSP: FOR J=1 TO CP: FOR K=1 TO C(J,1)
3060 FOR L=1 TO C(J,2): DIM A$(C(J,L*2+1)): PRINT PAPER C(J,L*2+2);A$;: NEXT L: NEXT K
3070 NEXT J: NEXT I: NEXT H
3080 READ ND: IF NOT ND THEN RETURN
3090 FOR H=1 TO ND: READ CI,CD,CX,CY: GO SUB 3270: IF NOT CD THEN GO TO 3250
3100 READ T: IF CD>3 THEN GO TO 3150
3110 LET F=1: IF CD=2 THEN FOR I=T TO T-2 STEP -1: PLOT CX,CY: DRAW I/2,I: DRAW I/2,-I: DRAW -I,0: PLOT CX,CY+I*.66: DRAW I,0: DRAW -I/2,-I: DRAW -I/2,I: LET CX=CX+1: LET CY=CY+.66: NEXT I: GO TO 3260
3120 LET A=CD+4: LET B=180/A: LET C=B+90: LET D=C+180: LET E=B-90
3130 FOR T=T TO F STEP -1: PLOT CX,CY: FOR I=1 TO A: LET E=E+C: DRAW SIN (E*PI/180)*T,COS (E*PI/180)*T: LET E=E+D: DRAW SIN (E*PI/180)*T,COS (E*PI/180)*T: NEXT I: LET E=E+C: DRAW SIN (E*PI/180)*T,COS (E*PI/180)*T
3140 LET E=B-90: LET CX=CX+45/B: NEXT T: GO TO 3260
3150 IF CD=4 THEN FOR I=1 TO T: CIRCLE CX,CY,I: NEXT I: FOR I=10 TO 360 STEP 20: PLOT CX,CY: LET A=I*PI/180: DRAW (SIN A)*T*1.5,(COS A)*T*1.5: NEXT I: GO TO 3260
3160 IF CD=5 THEN FOR I=T TO 1 STEP -2: PLOT CX,CY: DRAW 0,I: LET CX=CX+1: LET CY=CY+1: NEXT I: GO TO 3260
3170 IF CD=6 THEN FOR I=T TO 1 STEP -2: PLOT CX,CY: DRAW 0,I: LET CX=CX-1: LET CY=CY+1: NEXT I: GO TO 3260
3180 IF CD>9 THEN GO TO 3220
3190 LET A=0: LET B=T: IF CD=8 THEN LET A=-T
3200 IF CD=9 THEN LET A=-T: LET B=0
3210 FOR I=PI TO PI+.85 STEP -.05: PLOT CX,CY: DRAW A,B,-I: NEXT I: GO TO 3260
3220 IF CD=10 THEN FOR I=1 TO 10: PLOT CX,CY: DRAW T,175-CY: LET CX=CX+1: NEXT I: GO TO 3260
3230 IF CD=11 THEN FOR I=1 TO 10: PLOT CX,CY: DRAW -T,175-CY: LET CX=CX-1: NEXT I: GO TO 3260
3235 IF CD=12 THEN FOR I=1 TO T: CIRCLE CX,CY,I: NEXT I: GO TO 3260
3240 IF CD=13 THEN CIRCLE CX,CY,T: GO TO 3260
3245 IF CD=14 THEN PLOT CX,CY: DRAW T/2,T: DRAW T/2,-T: DRAW -T,0: GO TO 3260
3247 IF CD=15 THEN LET F=T-3: LET CD=1: GO TO 3120
3250 READ T: FOR I=1 TO T: READ E$: PRINT AT CY,CX;E$: LET CY=CY+1: NEXT I
3260 NEXT H: RETURN
3270 LET O=0: IF CI>63 THEN LET O=1: LET CI=CI-64
3280 LET PR=INT (CI/8): LET I=CI-PR*8: OVER O: PAPER PR: INK I: RETURN
5001 DATA 630
5002 DATA 60
5003 DATA 130
5004 DATA 220
5010 DATA "AMERICA","EUROPA","AFRICA","ASIA","OCEANIA"
5100 DATA "AMERICA DEL NORTE",4,6000
5101 DATA "AMERICA CENTRAL",11,6004
5102 DATA "AMERICA ANTILLANA",17,6015
5103 DATA "AMERICA DEL SUR",14,6032
5104 DATA "EUROPA OCCIDENTAL",24,6046
5105 DATA "EUROPA ORIENTAL",10,6070
5106 DATA "AFRICA DEL NORTE",13,6080
5107 DATA "AFRICA OCCIDENTAL",19,6093
5108 DATA "AFRICA DEL SUR",23,6112
5109 DATA "ASIA SUDOCCIDENTAL",16,6135
5110 DATA "ASIA CENTRAL",18,6151
5111 DATA "ASIA SUDORIENTAL",13,6169
5112 DATA "OCEANIA",7,6182
6000 DATA "GROENLANDIA","GODTHAAB",1,9
6001 DATA "CANADA","OTTAWA",1,1,1,1,1,21,3,8,2,16,7,8,2,1,58,0,10,4,12,"...","...","...","..."
6002 DATA "ESTADOS UNIDOS","WASHINGTON",1,4,2,6,2,1,1,32,2,1,1,32,7,1,2,1,1,32,2,6,1,32,4,1,15,0,0,0,7," * * * * * "," * * * * "," * * * * * "," * * * * "," * * * * * "," * * * * "," * * * * * "
6003 DATA "MEXICO","MEXICO",1,9
6004 DATA "GUATEMALA","GUATEMALA",1,4,1,1,1,21,3,11,1,10,7,11,1,2,60,0,12,7,8,"...","...","...","...","...","...","...",62,0,15,9,5,"...","...","...","...","..."
6005 DATA "HONDURAS","TEGUCIGALPA",1,4,1,1,3,7,1,32,1,7,1,32,7,7,1,32,1,5,57,1,100,100,5,57,1,140,100,5,57,1,120,90,5,57,1,100,80,5,57,1,140,80,5
6006 DATA "EL SALVADOR","SAN SALVADOR",1,9
6007 DATA "NICARAGUA","MANAGUA",1,4,1,1,3,7,1,32,1,7,1,32,7,7,1,32,1,2,57,14,112,76,30,57,13,127,87,20
6008 DATA "COSTA RICA","SAN JOSE",1,4,1,1,5,4,1,32,1,3,1,32,7,7,1,32,2,3,1,32,7,4,1,32,1,1,20,12,60,91,16
6009 DATA "PANAMA","PANAMA",1,4,1,1,3,10,2,16,7,16,2,10,2,16,1,16,7,1,1,32,4,2,57,1,45,140,12,58,1,172,54,12
6010 DATA "CUBA","LA HABANA",1,4,1,1,6,4,1,32,1,4,1,32,7,4,2,7,7,25,1,4,1,32,7,4,1,32,1,1,1,32,4,2,10,5,0,16,159,122,1,25,96,10
6011 DATA "JAMAICA","KINGSTON",1,9
6012 DATA "HAITI","PUERTO PRINCIPE",1,9
6013 DATA "REP.DOMINICANA","SANTO DOMINGO",1,4,1,1,3,9,3,14,1,4,7,14,2,3,1,32,7,9,3,14,2,4,7,14,1,0
6014 DATA "PUERTO RICO","SAN JUAN",1,4,1,1,6,4,1,32,2,4,1,32,7,4,2,7,7,25,2,4,1,32,7,4,1,32,2,1,1,32,4,2,9,5,0,16,159,121,1,25,96,10
6015 DATA "BELICE","BELMOPAN",1,9
6016 DATA "BAHAMAS","NASSAU",1,4,1,1,3,7,1,32,5,7,1,32,6,7,1,32,5,1,0,5,0,8,167
6017 DATA "ANGUILLA","THE VALLEY",1,9
6018 DATA "CAIMAN","GEORGETOWN",1,9
6019 DATA "I.VIRGENES-USA","CHARLOTTE AMALIE",1,9
6020 DATA "ISLAS VIRGENES","ROAD TOWN",1,9
6021 DATA "ST.KITTS","BASSETERRE",1,9
6022 DATA "ANTIGUA","SAINT JOHN'S",1,9
6023 DATA "MONTSERRAT","PLYMOUTH",1,9
6024 DATA "GUADALUPE","BASSE TERRE",1,9
6025 DATA "DOMINICA","ROSEAU",1,9
6026 DATA "MARTINICA","FORT DE FRANCE",1,9
6027 DATA "SANTA LUCIA","CASTRIES",1,9
6028 DATA "SAN VICENTE","KINGSTOWN",1,9
6029 DATA "BARBADOS","BRIDGETOWN",1,4,1,1,1,21,3,11,1,10,6,11,1,1,48,0,13,6,9,"...","...","...","...","...","...","...","...","...","..."
6030 DATA "GRANADA","SAINT GEORGE'S",1,9
6031 DATA "ANTILLAS HOLAND","WILLEMSTAD",1,9
6032 DATA "GUAYANA FRANCES","CAYENA",1,9
6033 DATA "SURINAM","PARAMARIBO",1,1,1,1,5,4,1,32,4,3,1,32,7,7,1,32,2,3,1,32,7,4,1,32,4,1,22,1,115,91,12
6034 DATA "GUYANA","GEORGETOWN",1,9
6035 DATA "VENEZUELA","CARACAS",1,4,1,1,3,7,1,32,6,7,1,32,1,7,1,32,2,2,52,12,24,147,16,15,0,10,8,6,"       *","    *    *","","   *   *","","*      *"
6036 DATA "TRINIDAD-TOBAGO","PUERTO ESPAÑA",1,1,1,1,1,21,1,32,2,3,0,11,255,8,225,0,11,245,8,225,0,11,235,8,225
6037 DATA "COLOMBIA","BOGOTA",1,4,1,1,3,11,1,32,6,5,1,32,1,5,1,32,2,0
6038 DATA "ECUADOR","QUITO",1,9
6039 DATA "PERU","LIMA",1,1,1,1,1,21,3,11,2,10,7,11,2,1,60,0,12,7,7,"...","...","...","...","...","...","..."
6040 DATA "BOLIVIA","SUCRE",1,1,1,1,3,7,1,32,2,7,1,32,6,7,1,32,4,0
6041 DATA "PARAGUAY","ASUNCION",1,4,1,1,3,7,1,32,2,7,1,32,7,7,1,32,1,2,57,13,127,91,16,57,13,127,91,10
6042 DATA "BRASIL","BRASILIA",1,1,1,1,3,6,1,32,4,9,3,11,4,10,1,11,4,6,1,32,4,3,62,6,127,16,151,62,5,128,16,151,78,12,127,91,38
6043 DATA "URUGUAY","MONTEVIDEO",1,4,3,2,2,2,1,32,7,2,2,10,7,22,5,2,2,2,1,32,7,2,1,32,5,1,2,2,1,32,7,3,1,32,4,1,62,4,40,135,20
6044 DATA "ARGENTINA","BUENOS AIRES",1,4,1,1,3,7,1,32,5,7,1,32,7,7,1,32,5,1,62,4,127,91,16
6045 DATA "CHILE","SANTIAGO",1,4,1,1,2,10,2,10,1,22,7,11,1,32,2,1,15,1,27,138,10
6046 DATA "PORTUGAL","LISBOA",2,9
6047 DATA "ESPAÑA","MADRID",2,1,1,1,3,5,1,32,2,11,1,32,6,5,1,32,2,0
6048 DATA "ANDORRA","ANDORRA LA VIEJA",2,4,1,1,1,21,3,11,1,10,6,11,2,1,50,0,13,7,6,"...","..."
```

```
[redacted]
6049 DATA "FRANCIA","PARIS",2,4,
1,1,1,21,3,11,1,10,7,11,2,0
6050 DATA "GRAN BRETAÑA","LONDRE
S",2,4,1,1,5,8,5,13,1,1,7,4,2,1,
7,13,1,1,3,14,7,4,2,14,7,3,1,32,
2,1,3,14,7,4,2,14,7,8,5,13,1,1,7
,4,2,1,7,13,1,2,10,10,0,8,245,10
,11,255,8,245
6051 DATA "IRLANDA","DUBLIN",2,1
,1,1,1,21,3,11,4,10,7,11,6,0
6052 DATA "ISLANDIA","REYKJAVIK"
,2,4,1,1,5,8,5,8,1,1,7,3,2,1,7,1
9,1,1,3,9,7,3,2,20,7,3,1,32,2,1,
3,9,7,3,2,20,7,8,5,8,1,1,7,3,2,1
,7,19,1,0
6053 DATA "BELGICA","BRUSELAS",2
,1,1,1,1,21,3,11,0,10,6,11,2,0
6054 DATA "HOLANDA","AMSTERDAM",
2,4,1,1,3,7,1,32,2,7,1,32,7,7,1,
32,1,0
6055 DATA "SUIZA","BERNA",2,1,1,
1,6,3,1,32,2,5,3,14,2,4,7,14,2,4
,3,9,2,14,7,9,2,5,3,14,2,4,7,14,
2,3,1,32,2,1,1,32,1,0
6056 DATA "ITALIA","ROMA",2,1,1,
1,1,21,3,11,4,10,7,11,2,0
6057 DATA "GRECIA","ATENAS",2,4,
1,1,4,8,3,14,5,4,7,14,5,4,1,32,7
,8,3,14,5,4,7,14,5,1,1,32,4,0
6058 DATA "VATICANO","",2,1,1,1,
1,21,2,16,6,16,7,0
6059 DATA "MONACO","MONACO",2,4,
1,1,3,10,1,32,2,10,1,32,7,1,1,32
,4,0
6060 DATA "SAN MARINO","SAN MARI
NO",2,4,1,1,3,10,1,32,7,10,1,32,
5,1,1,32,4,0
6061 DATA "LIECHTENSTEIN","VADUZ
",2,4,1,1,3,10,1,32,1,10,1,32,2,
1,1,32,4,1,14,0,3,3,4," [redacted] "," [redacted]
 [redacted] "," [redacted] "," [redacted] "
6062 DATA "AUSTRIA","VIENA",2,1,
1,1,3,7,1,32,2,7,1,32,7,7,1,32,2
,0
6063 DATA "LUXEMBURGO","LUXEMBUR
GO",2,4,1,1,3,7,1,32,2,7,1,32,7,
7,1,32,5,0
6064 DATA "REP.FED.ALEMANA","BON
N",2,1,1,1,3,7,1,32,0,7,1,32,2,7
,1,32,6,0
6065 DATA "DINAMARCA","COPENHAGU
E",2,1,1,1,4,9,3,9,2,2,7,21,2,2,
1,32,7,9,3,9,2,2,7,21,2,1,1,32,1
,0
6066 DATA "NORUEGA","OSLO",2,4,1
,1,5,8,5,10,2,1,7,3,1,1,7,17,2,1
,3,11,7,3,1,18,7,3,1,32,1,1,3,11
;7,3,1,18,7,8,5,10,2,1,7,3,1,1,7
,17,2,0
6067 DATA "SUECIA","ESTOCOLMO",2
,4,1,1,3,9,3,11,1,3,6,18,1,3,1,3
2,6,9,3,11,1,3,6,18,1,0
6068 DATA "FINLANDIA","HELSINKI"
,2,1,1,1,3,8,3,8,7,5,5,19,7,5,1,
32,5,8,3,8,7,5,5,19,7,0
6069 DATA "MALTA","VALLETTA",2,1
,1,1,1,21,2,16,7,16,2,1,56,0,1,1
,3," [redacted] "," [redacted] "," [redacted] "
6070 DATA "REP.POP.ALEMANA","BER
LIN",2,1,1,1,3,7,1,32,0,7,1,32,2
,7,1,32,6,3,60,13,127,91,32,60,1
3,127,91,36,22,0,13,7,7," [redacted] ",
" [redacted] "," [redacted] "," [redacted] "," [redacted]
"," [redacted] "," [redacted] "
6071 DATA "POLONIA","VARSOVIA",2
,1,1,1,3,10,1,32,7,10,1,32,2,1,1
,32,1,0
6072 DATA "CHECOSLOVAQUIA","PRAG
A",2,4,1,1,3,10,1,32,7,10,1,32,2
,1,1,32,4,1,57,5,0,16,159
6073 DATA "HUNGRIA","BUDAPEST",2
,1,1,1,3,7,1,32,2,7,1,32,7,7,1,3
2,4,0
6074 DATA "YUGOSLAVIA","BELGRADO
",2,4,1,1,3,7,1,32,1,7,1,32,7,7,
1,32,2,1,58,1,100,93,20
6075 DATA "ALBANIA","TIRANA",2,9
6076 DATA "TURQUIA EUROPEA","EST
AMBUL",2,9
6077 DATA "BULGARIA","SOFIA",2,1
,1,1,3,7,1,32,7,7,1,32,4,7,1,32,
2,1,61,12,24,151,12
6078 DATA "RUMANIA","BUCAREST",2
,9
6079 DATA "UNION SOVIETICA","MOS
CU",2,1,1,1,1,21,1,32,2,2,22,1,3
2,159,5,22,0,4,3,4," [redacted] "," [redacted] "
," [redacted] "," [redacted] "
6080 DATA "MAURITANIA","NUAKCHOT
T",3,1,1,1,1,21,1,32,4,2,38,1,11
5,100,10,38,9,160,91,60
6081 DATA "MARRUECOS","RABAT",3,
1,1,1,1,21,1,32,2,1,20,15,110,91
,15
6082 DATA "ARGELIA","ARGEL",3,1,
1,1,1,21,2,16,4,16,7,2,34,7,130,
61,60,58,1,125,91,10
6083 DATA "TUNEZ","TUNEZ",3,9
6084 DATA "LIBIA","TRIPOLI",3,1,
1,1,3,7,1,32,2,7,1,32,7,7,1,32,0
,0
6085 DATA "EGIPTO","EL CAIRO",3,
4,1,1,3,7,1,32,2,7,1,32,7,7,1,32
,0,1,62,0,13,8,5," [redacted] "," [redacted] "
," [redacted] "," [redacted] "," [redacted] "
6086 DATA "SOMALIA","MOGADISCIO"
,3,9
6087 DATA "YIBUTI","YIBUTI",3,9
6088 DATA "ETIOPIA","ADDIS ABEBA
",3,1,1,1,3,7,1,32,4,7,1,32,6,7,
1,32,2,0
6089 DATA "SUDAN","JARTUM",3,1,1
,1,3,7,1,32,2,7,1,32,7,7,1,32,0,
1,4,5,0,8,167
6090 DATA "CHAD","N'DJAMENA",3,4
,1,1,1,21,3,11,1,10,6,11,2,0
6091 DATA "NIGER","NIAMEY",3,1,1
,1,3,7,1,32,6,7,1,32,7,7,1,32,4,
1,62,12,127,91,16
6092 DATA "MALI","BAMAKO",3,9
6093 DATA "ISLAS CANARIAS","LAS
PALMAS",3,9
6094 DATA "SENEGAL","DAKAR",3,1,
1,1,1,21,3,11,4,10,6,11,2,1,52,1
,115,91,10
6095 DATA "GAMBIA","BANJUL",3,1,
1,1,5,7,1,32,2,1,1,32,7,5,1,32,5
,1,1,32,7,7,1,32,4,0
6096 DATA "GUINEA BISSAU","BISSA
U",3,1,1,1,3,10,2,11,2,21,6,10,2
,11,2,21,4,1,1,32,1,1,48,1,16,95
,15
6097 DATA "GUINEA","CONAKRY",3,1
,1,1,1,21,3,11,2,10,6,11,4,0
6098 DATA "SIERRA LEONA","FREETO
UN",3,3,1,1,3,7,1,32,4,7,1,32,7,
7,1,32,1,0
6099 DATA "LIBERIA","MONROVIA",3
,4,3,2,2,2,2,9,1,23,2,2,2,9,1,23
,7,1,1,1,2,9,1,23,2,3,2,2,1,32,7
,2,1,32,2,1,15,1,24,139,10
6100 DATA "COSTA DE MARFIL","ABI
DJAN",3,1,1,1,1,21,3,11,3,10,7,1
1,4,0
6101 DATA "ALTO VOLTA","UAGADUGU
",3,1,1,1,3,7,1,32,0,7,1,32,7,7,
1,32,2,0
6102 DATA "GHANA","ACCRA",3,1,1,
1,3,7,1,32,2,7,1,32,6,7,1,32,4,1
,48,1,110,91,15
6103 DATA "TOGO","LOME",3,1,1,1,
6,4,2,12,2,20,4,4,2,12,2,20,6,4,
2,12,2,20,4,4,1,32,6,4,1,32,4,1,
1,32,1,1,23,1,28,130,15
6104 DATA "DAHOMEY","PORTO NOVO"
,3,1,1,1,1,21,1,32,4,1,34,1,28,1
30,15
6105 DATA "NIGERIA","LAGOS",3,1,
1,1,1,21,3,11,4,10,7,11,4,0
6106 DATA "CAMERUN","YAUNDE",3,1
,1,1,1,21,3,11,4,10,2,11,6,1,22,
1,115,91,10
6107 DATA "REP.CENTROAFRIC","BAN
GUI",3,9
6108 DATA "CABO VERDE","PRAIA",3
,1,1,1,3,10,2,10,2,22,6,10,2,10,
2,22,4,1,1,32,1,2,16,1,28,147,10
,20,13,42,145,18
6109 DATA "GUINEA ECUATOR.","MAL
ABO",3,1,1,1,3,7,1,32,4,7,1,32,7
,7,1,32,2,1,5,5,0,8,167
6110 DATA "GABON","LIBREVILLE",3
,9
6111 DATA "CONGO","BRAZZAVILLE",
3,9
6112 DATA "ZAIRE","KINSHASA",3,9
6113 DATA "ANGOLA","LUANDA",3,9
6114 DATA "NAMIBIA","WINDHUK",3,
9
6115 DATA "REP.SUDAFRICANA","PRE
TORIA",3,9
6116 DATA "TRANSKEI","UMTATA",3,
9
6117 DATA "LESOTHO","MASERU",3,1
,1,1,1,21,3,5,4,5,2,22,5,0
6118 DATA "SUAZILANDIA","MBABANE
",3,4,1,1,5,4,1,32,1,1,1,32,6,11
,1,32,2,1,1,32,6,4,1,32,1,0
6119 DATA "MOZAMBIQUE","MAPUTO",
3,9
6120 DATA "VENDA","THOHOYANDOU",
3,9
6121 DATA "BOPHUTHATSWANA","MMAB
ATHO",3,9
6122 DATA "BOTSWANA","GABERONES"
,3,1,1,1,5,7,1,32,5,1,1,32,7,5,1
,32,0,1,1,32,7,7,1,32,5,0
6123 DATA "ZIMBAWE","SALISBURY",
3,9
6124 DATA "ZAMBIA","LUSAKA",3,1,
1,1,2,7,1,32,4,14,4,20,4,4,2,4,0
,4,3,0
6125 DATA "MALAWI","LILONGWE",3,
9
6126 DATA "TANZANIA","DAR-ES-SAL
AAM",3,9
6127 DATA "UGANDA","KAMPALA",3,1
,2,2,3,3,1,32,0,3,1,32,6,3,1,32,
2,1,1,3,1,32,1,1,7,12,127,103,12
6128 DATA "KENIA","NAIROBI",3,1,
1,1,6,6,1,32,0,1,1,32,7,6,1,32,2
,1,1,32,7,6,1,32,4,1,1,32,1,0
6129 DATA "RUANDA","KIGALI",3,9
6130 DATA "BURUNDI","BUJUMBURA",
3,9
6131 DATA "MADAGASCAR","ANTANANA
RIVO",3,1,1,1,3,10,2,10,7,22,2,1
0,2,10,7,22,4,1,1,32,1,0
6132 DATA "COMORES","MORONI",3,9
6133 DATA "MAURICIO","PORT LOUIS
",3,1,1,1,5,5,1,32,2,5,1,32,5,5,
1,32,6,5,1,32,4,1,1,32,1,0
6134 DATA "SEYCHELLES","VICTORIA
",3,9
6135 DATA "TURQUIA","ANKARA",4,9
6136 DATA "CHIPRE","NICOSIA",4,9
6137 DATA "SIRIA","DAMASCO",4,9
6138 DATA "LIBANO","BEIRUT",4,9
6139 DATA "ISRAEL","JERUSALEM",4
,4,1,1,6,2,1,32,7,2,1,32,1,12,1,
32,7,2,1,32,1,2,1,32,7,1,1,32,4,
1,57,2,107,83,40
6140 DATA "JORDANIA","AMMAN",4,1
,1,1,3,7,1,32,0,7,1,32,7,7,1,32,
4,2,58,3,24,91,5,66,5,0,8,167
6141 DATA "EGIPTO","EL CAIRO",4,
9
6142 DATA "ARABIA SAUDITA","RIYA
DH",4,9
6143 DATA "YEMEN","SAN'A",4,1,1,
1,3,7,1,32,2,7,1,32,7,7,1,32,0,1
,4,1,115,91,10
6144 DATA "R.D.P.YEMEN","ASH SHA
'AB",4,1,1,1,3,7,1,32,2,7,2,7,2,
25,7,7,1,32,0,2,85,5,0,8,167,85,
1,24,91,10
6145 DATA "OMAN","MASCATE",4,9
6146 DATA "EMIR.ARABES UNID","AB
U DHABI",4,1,1,1,3,7,2,11,2,21,4
,7,2,11,2,21,7,7,2,11,2,21,0,0
6147 DATA "QATAR","DOHA",4,9
6148 DATA "BAHREIN","MANAMA",4,9
6149 DATA "KUWAIT","AL KUWAIT",4
,1,1,1,3,7,1,32,4,7,2,7,0,25,7,7
,1,32,2,2,0,5,0,8,112,0,5,0,63,1
12
6150 DATA "IRAK","BAGDAD",4,1,1,
1,3,7,1,32,2,7,1,32,7,7,1,32,0,3
,4,1,80,91,10,4,1,115,91,10,4,1,
150,91,10
6151 DATA "IRAN","TEHERAN",4,9
6152 DATA "AFGANISTAN","KABUL",4
,9
6153 DATA "PAKISTAN","ISLAMABAD"
,4,1,1,1,1,21,2,10,7,22,4,2,7,8,
210,60,64,7,1,185,110,7
6154 DATA "INDIA","NUEVA DELHI",
4,9
6155 DATA "NEPAL","KATMANDU",4,9
6156 DATA "BHUTAN","PUNAKHA",4,9
6157 DATA "BANGLADESH","DACCA",4
,1,1,1,1,21,1,32,0,1,2,12,100,91
,40
6158 DATA "SRI LANKA","COLOMBO",
4,9
6159 DATA "CHINA","PEKIN",4,1,1,
1,1,21,1,32,2,2,6,1,24,140,10,22
,0,8,1,7,"*",""," *",""," *","
","*"
6160 DATA "HONG KONG","VICTORIA"
,4,9
6161 DATA "MACAO","",4,9
6162 DATA "TAIWAN","TAIPEI",4,9
6163 DATA "ISLAS MALDIVAS","MALE
",4,9
6164 DATA "JAPON","TOKYO",4,1,0,
1,2,12,127,91,48
6165 DATA "COREA DEL SUR","SEUL"
,4,9
6166 DATA "COREA DEL NORTE","PYA
NGYANG",4,9
6167 DATA "MONGOLIA","ULAN BATOR
",4,9
6168 DATA "UNION SOVIETICA","MOS
CU",4,9
6169 DATA "BIRMANIA","RANGUN",4,
9
6170 DATA "TAILANDIA","BANGKOK",
4,4,1,1,5,4,1,32,2,3,1,32,7,7,1,
32,1,3,1,32,7,4,1,32,2,0
6171 DATA "LAOS","VIENTIANE",4,4
,1,1,3,5,1,32,2,11,1,32,1,5,1,32
,2,1,7,12,127,91,40
6172 DATA "CAMBOYA","PHNOM PENH"
,4,9
6173 DATA "VIETNAM D.NORTE","HAN
OI",4,9
6174 DATA "VIETNAM DEL SUR","SAI
GON",4,9
6175 DATA "MALASIA","KUALA LUMPU
R",4,9
6176 DATA "SINGAPUR","SINGAPUR",
4,1,1,1,1,11,1,32,2,2,7,7,72,90,
80,23,0,7,3,4,"  *","*   *","","
 * *"
6177 DATA "BRUNEI","BANDAR S.BEW
AGAN",4,9
6178 DATA "INDONESIA","YAKARTA",
4,9
6179 DATA "FILIPINAS","CIUDAD QU
EZON",4,9
6180 DATA "SIKKIM","GANGTOK",4,9
6181 DATA "TIMOR","DILLI",4,9
6182 DATA "AUSTRALIA","CANBERRA"
,5,9
6183 DATA "FIDJI","SUVA",5,9
6184 DATA "HAWAII","HONOLULU",5,
9
6185 DATA "NUEVA ZELANDIA","WELL
INGTON",5,9
6186 DATA "SAMOA OCCIDENT.","API
A",5,9
6187 DATA "TONGA","NUKUALOTA",5,
9
6188 DATA "NUEVA GUINEA","RABAUL
",5,9
```

HUESOS



COMP.: TI99/4A
CLAS.: EDU



Nuestra computadora nos mostrará con excelentes gráficos los huesos de todo el esqueleto humano, y, durante el desarrollo del programa, nos irá evaluando los conocimientos. K64

```
115 REM ********************
116 REM *  H U E S O S   *
117 REM ********************
150 GOTO 2480
160 FOR I=1 TO N
170 READ C,C$
180 CALL CHAR(C,C$)
190 NEXT I
200 CALL CLEAR
210 RETURN
220 DATA 79,80,82,73,77,65,32,60,69,78,84,69,82,62,32
230 RESTORE 220
240 FOR Y=18 TO 32
250 READ G
260 CALL HCHAR(24,Y,G)
270 NEXT Y
280 CALL KEY(0,K,S)
290 IF K<>13 THEN 280
300 CALL HCHAR(24,18,32,14)
310 RETURN
320 FOR I=1 TO N
330 READ X,Y,G,R
340 CALL HCHAR(X,Y,G,R)
350 NEXT I
360 RETURN
370 FOR I=1 TO R
380 READ BONE$(I),B(I)
390 NEXT I
400 RANDOMIZE
410 FOR I=1 TO R
420 RR=INT(RND*R+1)
430 IF BONE$(RR)="" THEN 420
440 B$(RR)=BONE$(RR)
450 BB(RR)=B(RR)
460 ANS(RR)=I
470 CALL HCHAR(15+I,2,48+I)
480 FOR J=1 TO LEN(B$(RR))
490 CALL HCHAR(15+I,J+3,ASC(SEG$(B$(RR),J,I)))
500 NEXT J
510 BONE$(RR)=""
520 NEXT I
530 DATA 78,79,77,66,82,69,32,69,83,84,69,32,72,85,69,83,79
540 RESTORE 530
550 FOR Y=9 TO 25
560 READ G
570 CALL HCHAR(1,Y,G)
580 NEXT Y
590 FOR I=1 TO R
600 RR=INT(RND*R+1)
610 IF B$(RR)="" THEN 600
620 CALL HCHAR(14,2,63,3)
630 CALL KEY(0,K,S)
640 CALL COLOR(BB(RR),16,1)
650 CALL COLOR(BB(RR),7,1)
660 IF S<1 THEN 630
670 IF K-48=ANS(RR)THEN 700
680 CALL SOUND(500,-5,1)
690 GOTO 630
700 CALL HCHAR(14,2,32,3)
710 CALL SOUND(150,262,1)
720 CALL SOUND(150,330,1)
730 CALL SOUND(150,392,1)
740 CALL SOUND(150,330,1)
750 CALL SOUND(200,262,1)
760 CALL COLOR(BB(RR),12,1)
770 B$(RR)=""
780 NEXT I
790 DATA 78,85,69,86,65,32,80,82,85,69,66,65
800 RESTORE 790
810 FOR Y=20 TO 31
820 READ G
830 CALL HCHAR(23,Y,G)
840 NEXT Y
850 CALL HCHAR(24,26,40)
860 CALL HCHAR(24,27,83)
870 CALL HCHAR(24,28,47)
880 CALL HCHAR(24,29,78)
890 CALL HCHAR(24,30,41)
900 CALL KEY(0,K,S)
910 IF K=78 THEN 2480
920 IF K<>83 THEN 900
930 FOR Y=16 TO 24
940 CALL HCHAR(Y,2,32,12)
950 NEXT Y
960 CALL HCHAR(23,19,32,13)
970 CALL HCHAR(24,26,32,5)
980 RETURN
990 RESTORE 1000
1000 DATA 97,F0FCFFFFFFFFFFFF,98,0000C0E0F8FCFEFF,99,0080C0C0E0E0E0E,100,E0E0F0F
0F8F8FCFC
1010 DATA 101,FCFCFEFE3F1F0F07,102,FFFFFFFFFEF8F0C,103,FFFF7F1F1F0F0707,109,0F7F
FFFFFFFFFFFF
1020 DATA 105,000000031F3F7FFF,106,01070F0F1F1F3F7F,107,7F7F3F3F3F3F3F1F,108,0F0
F0F0F0F0F0F0F
1030 DATA 137,0F0F0F0F03030101,138,FFFF7F7F3F1E0C,155,FFFEFCF,153,030303030303
03,159,F0C0808
1040 DATA 154,030303030303070F,156,0080C0C0E0E0F0F8,157,0F3FFFFFFFFFFFFE,158,000
00001030FFFFF
1050 DATA 113,7C7E7F7F3F3F1F1F,114,0F0FFFFFFFFFFFFF,115,1F1F0F0F0F0F1F1F,116,FFF
FFFFFFEE0C0C
1060 DATA 117,00000080E0FCFFFF,118,FFFFFFFFFF3F1F07,121,FF7F3F1F1F0F0701,122,7F1
F0701
1070 DATA 123,FFFFFFFF0F,124,FEFCF0C08,125,CFCF878703030101,126,0080E0F0F0F8F8F8
1080 DATA 127,00BBBBBBFFFFFFFF,145,FFFFFFFF77777777,146,FFFF7B7B7B7B,147,FFFFFFF
FFFDB1BC
1090 N=37
1100 GOSUB 160
1110 PRINT "PARIETAL";TAB(20);"FRONTAL":::::"TEMPORAL"::
1120 PRINT "CIGOMATICO";TAB(19);"MAXILAR":::"    MANDIBULA"::::::::
1130 RESTORE 1140
1140 DATA 3,14,105,1,3,15,109,1,3,16,96,1,3,17,97,1,3,18,98,1,4,13,106,1,4,14,10
4,1
1150 DATA 4,15,96,4,4,19,97,1,4,20,98,1,5,12,106,1,5,13,104,2,5,15,96,6,5,21,99,
1
1160 DATA 6,12,104,3,6,15,96,6,6,21,100,1,7,12,104,3,7,15,96,5,7,20,102,1,7,21,1
01,1
1170 DATA 8,12,107,1,8,13,136,3,8,16,155,1,8,18,103,1,8,19,96,1,8,21,153,1,9,12,
108,1
1180 DATA 9,13,136,3,9,18,154,1,9,19,152,1,9,20,156,1,9,21,153,1,10,12,108,1,10,
13,136,2
1190 DATA 10,15,112,1,10,16,117,1,10,17,158,1,10,18,157,1,10,19,159,1,10,20,113,
1,10,21,114,1
1200 DATA 11,12,137,1,11,13,136,1,11,14,120,1,11,15,118,1,11,16,144,3,11,20,115,
1,11,21,116,1
1210 DATA 12,13,138,1,12,14,125,1,12,15,126,1,12,16,144,5,13,15,120,1,13,16,147,
1,13,17,146,1
1220 DATA 13,18,145,3,14,15,121,1,14,16,120,1,14,17,127,4,15,16,121,1,15,17,120,
4
1230 DATA 16,17,122,1,16,18,123,2,16,20,124,1,16,21,32,1
1240 N=66
1250 GOSUB 320
1260 GOSUB 230
1270 RESTORE 1280
1280 DATA 6,3,32,8,6,22,32,7,11,3,32,8,13,3,32,10,13,21,32,7,16,7,32,9
1290 N=6
1300 GOSUB 320
1310 R=6
1320 DATA FRONTAL,9,PARIETAL,10,CIGOMATICO,11,MANDIBULA,12,TEMPORAL,14,MAXILAR,1
5
1330 RESTORE 1320
1340 GOSUB 370
1350 GOTO 1330
1360 RESTORE 1370
1370 DATA 97,0001030307070707,98,0F0F0F0F0F0F8F0F,99,0707070703030301,100,81C3C7EF
FFFFFFFF
1380 DATA 101,F7F3F0F0F0F3F7FF,102,FFFFFFFFF7F7E3C1,113,000000000FFFFFF,121,7E7
```

```
E7E7E007E7E7E
1390 DATA 129,FEFEFEFEFE000000,130,FEFEFEFEFE0000FE,131,FEFEFE0000FEFEFE,137,FFF
FFFFE000080C
1400 DATA 138,FFFFFFFFFF0000FF,139,80C0C0C0800000FE,140,FFFFFFFF0000FFFF,141,FFF
FFF0000FFFFFF
1410 DATA 142,FFFFFF0000C0E0E0,143,C0C08,145,000001070F1F7FFF,146,1878F8F8F0E0C0
8
1420 DATA 64,0000001F1010101,94,101010101010101,95,101010101F
1430 N=23
1440 GOSUB 160
1450 PRINT TAB(19);"  FALANGE":TAB(12);"RADIO"
1460 CALL HCHAR(22,30,83)
1470 PRINT :::TAB(10);"CUBITO  ^_METACARPO"
1480 CALL HCHAR(23,31,83)
1490 PRINT " HUMERO"::TAB(18);"_CARPO"::::::::::
1500 CALL VCHAR(12,20,94,2)
1510 RESTORE 1520
1520 DATA 8,2,97,1,9,2,98,1,10,2,99,1,8,3,96,8,9,3,96,8,10,3,96,8,8,11,100,1
1530 DATA 9,11,101,1,10,11,102,1,8,12,104,8,9,12,113,8,10,12,112,8,8,20,121,1,9,
20,121,1
1540 DATA 10,20,121,1,7,21,145,1,8,21,129,1,9,21,130,1,10,21,131,1,7,22,146,1,8,
22,138,2
1550 DATA 9,22,140,2,8,24,139,1,9,24,137,1,10,22,141,1,10,23,142,1,10,24,143,1,1
0,29,32,1
1560 N=27
1570 GOSUB 320
1580 GOSUB 230
1590 N=7
1600 RESTORE 1610
1610 DATA 6,23,32,9,7,14,32,6,7,23,32,1,11,12,32,20,12,4,32,19,13,20,32,1,14,20,
32,7
1620 GOSUB 320
1630 R=6
1640 DATA HUMERO,9,RADIO,10,CUBITO,11,CARPO,12,METACARPO,13,FALANGES,14
1650 RESTORE 1640
1660 GOSUB 370
1670 GOTO 1650
1680 RESTORE 1690
1690 DATA 43,FFFF00FFFFFFFFFF,33,FFFFFFFF1F1F1F0F,34,FFFFFFFFE0F8FCFE,35,FFFFFFF
F
1700 DATA 36,FFFFFFFF071F3F7F,37,FFFFFFFFF8F8F8F8,97,0707070707070707,100,E0E0E0
E0E0E0E0E
1710 DATA 98,00C0F0FCFEFFFFFF,101,00030F3F7FFFFFFF,99,FF1F0701,102,FFF8E08,105,3
FFFFFFF1F1FFFFF
1720 DATA 106,FFF7E7F7F7F7F7FF,107,FCFFFFFFF8F8FFFF,108,070707FFFFFF0101,109,E0E
0E0FFFFFF808
1730 DATA 113,000000FFFFFF,114,060F1F03C0C0FFFF,117,C0F0F8C00303FFFF,115,0000C0E
18F1F
1740 DATA 116,06060CF8F8F,119,6060301F1F0F,121,000103070F1F3F7F,125,0080C0E0F0F8
FCFE
1750 DATA 123,7F3F1F0F070301,127,FEFCF8F0E0C08,133,EFF7E7E7C381,132,F7EFE7E7C381
,145,7D413E3E1C08
1760 DATA 122,000000FFFFFFFFFF,137,7C7E3F3F0F0F0701,141,3E7EFCFCF0F0E08,138,0001
0101C7FFFFFF
1770 DATA 140,00808080E3FFFFFF,139,FFFFFFF7F7E3C1,129,FFC1DDF9F3E7CFC1,151,FFFFF
FC1FDFDE1FD
1780 DATA 130,00C1E1F7F7F7F7F7,134,008387EFEFEFEFEF,131,EFEFEFEFEFEFEFEF,135,F7F
7F7F7F7F7F7F7
1790 N=42
1800 GOSUB 160
1810 CALL COLOR(1,12,1)
1820 CALL COLOR(2,12,1)
1830 PRINT TAB(18);"CLAVICULA"::"1 ESTERNON";TAB(21);"ESCAPULA"::::
1840 PRINT ::TAB(21);"COSTILLA":::::TAB(16);"ESPINA":::" 2 SACRO"
1850 PRINT TAB(20);"ILIACO":" 3 COXIS"::TAB(18);"SINFISIS":::
1860 RESTORE 1880
1870 CALL VCHAR(2,17,43,14)
1880 DATA 4,11,33,1,4,12,34,1,4,13,35,3,4,16,36,1,4,18,34,1,4,19,35,3,4,22,36,1,
4,23,37,1
1890 DATA 5,11,97,1,5,12,96,1,5,13,98,1,5,15,114,1,5,16,105,1,5,17,106,1,5,18,10
7,1,5,19,117,1
1900 DATA 5,21,101,1,5,22,96,1,5,23,100,1,6,11,97,1,6,12,96,2,6,14,113,2,6,16,10
8,1
1910 DATA 6,17,104,1,6,18,109,1,6,19,113,2,6,21,96,2,6,23,100,1,7,12,99,1,7,13,1
13,9
1920 DATA 7,17,104,1,7,22,102,1,8,12,113,11,8,17,104,1,9,12,113,4,9,16,116,1,9,1
8,119,1,9,19,113,4
1930 DATA 10,11,113,5,10,19,113,5,11,11,113,4,11,15,116,1,11,19,119,1,11,20,113,
4,12,10,113,5
1940 DATA 12,20,113,5,13,10,113,4,13,14,116,1,13,20,119,1,13,21,113,4,16,13,121,
1950 DATA 16,16,130,1,16,17,128,1,16,18,134,1,16,19,120,2,16,21,125,1,17,13,120,
3,17,16,131,1
1960 DATA 17,17,129,1,17,18,135,1,17,19,120,3,18,13,120,3,18,16,133,1,18,17,151,
1,18,18,132,1
1970 DATA 18,19,120,3,19,13,123,1,19,14,120,2,19,17,145,1,19,19,120,2,19,21,127,
1,20,14,123,1
1980 DATA 20,15,120,1,20,16,122,3,20,19,120,1
1990 DATA 20,20,127,1,21,15,137,1,21,16,138,1,21,17,139,1,21,18,140,1,21,19,141,
1
2000 N=82
2010 GOSUB 320
2020 GOSUB 230
2030 RESTORE 2040
2040 DATA 1,20,32,9,3,3,32,10,3,23,32,8,9,23,32,8,14,18,32,6,18,22,32,6,17,3,32,
10
2050 DATA 19,3,32,8,21,20,32,8,5,17,104,1,17,17,128,1,18,17,128,1,18,17,128,1
2060 N=13
2070 GOSUB 320
2080 CALL CHAR(145,"7F7F3E3E1C08")
2090 DATA CLAVICULA,1,ESPINA,2,ESCAPULA,9,ESTERNON,10,COSTILLA,11,ILIACO,12,SACR
O,13,COXIS,15,SINFISIS,14
2100 R=9
2110 RESTORE 2090
2120 GOSUB 370
2130 CALL VCHAR(16,13,121)
2140 CALL VCHAR(17,13,120,2)
2150 CALL VCHAR(19,13,123)
2160 GOTO 2110
2480 CALL CLEAR
2490 PRINT TAB(5);"NOMBRES DE HUESOS":::::::::::::::::::::
2500 A$="FFFFFFFFFFFFFFFF"
2510 CALL CHAR(96,A$)
2520 CALL COLOR(9,7,1)
2530 CALL HCHAR(7,15,96,3)
2540 CALL HCHAR(8,15,96,3)
2550 CALL HCHAR(9,15,96,3)
2560 CALL VCHAR(10,16,96,6)
2570 CALL HCHAR(11,13,96,7)
2580 CALL VCHAR(15,15,96,6)
2590 CALL VCHAR(15,17,96,6)
2600 CALL COLOR(2,2,1)
2610 IF FLAG=2 THEN 2820
2620 FOR I=1 TO 7
2630 CALL COLOR(9+I,12,1)
2640 CALL CHAR(96+8*I,A$)
2650 NEXT I
2660 FLAG=2
2670 PRINT "INSTRUCCIONES?(S/N)"
2680 CALL KEY(0,K,S)
2690 IF K=78 THEN 2820
2700 IF K<>83 THEN 2680
2710 CALL CLEAR
2720 PRINT "UD PUEDE ESTUDIAR NOMBRES DE":"HUESOS TODO EL TIEMPO QUE "
2730 PRINT "DESEE.LUEGO OPRIMA <ENTER>"
2740 PRINT "AL BORRARSE ESTA LEYENDA":"SERA SU TURNO DE ELEGIR"
2750 PRINT :"EL HUESO.ELIJA EL "::"NUMERO CORRECTO"
2760 PRINT ::"DEBE NOMBRAR EL HUESO"::"CORRECTAMENTE PARA CONTINUAR"::
2770 GOSUB 230
2780 FLAG=2
2790 GOTO 2480
2800 DATA 69,76,73,74,65,58,32,32,49,32,67,65,66,69,90,65,50,32,66,82,65,90,79,3
2
2810 DATA 51,32,84,79,82,83,79,32,52,32,70,73,78,32,32,32
2820 RESTORE 2800
2830 CALL HCHAR(23,1,32,21)
2840 FOR X=7 TO 15 STEP 2
2850 FOR Y=23 TO 30
2860 READ G
2870 CALL HCHAR(X,Y,G)
2880 NEXT Y
2890 NEXT X
2900 CALL KEY(0,K,S)
2910 IF S<1 THEN 2900
2920 IF K=52 THEN 2980
2930 IF (K>52)+(K<49)=-1 THEN 2900
2940 CALL CLEAR
2950 PRINT "UN MOMENTO POR FAVOR..":::
2960 CALL COLOR(9,12,1)
2970 ON K-48 GOTO 990,1360,1680
2980 CALL CLEAR
2990 END
```

RELOJ

Autor: Fabián Jofré
Club de Usuario MSX
Comentario: Hugo D. Caro
Roberto N. Tokuda
CLAS.: UTI



Cuando programamos, el tiempo se nos pasa volando.
Sucede que nos es muy difícil apartar la vista de la pantalla y fijarnos en el reloj pulsera...
Para que al menos veamos en qué parte del día nos encontramos, este programa nos muestra un relojito digital en pantalla constantemente (en el borde superior derecho).
El único detalle es que cuando carguemos programas en cassette o diskette, el reloj se detiene hasta concluir la operación.
Para utilizarlo, debemos cargar el listado fuente (si poseemos un ensamblador Z80) o el programa objeto (listado 2) con el cargador que detallamos a continuación. Finalmente deberemos cargar el programa BASIC adjunto, que nos permite poner en hora el reloj.

Programa Cargador

(Listado 2)

Si no poseemos un ensamblador para Z80, el programa assembly que presentamos a continuación puede cargarse en código de máquina.
Este listado muestra la posición de memoria en donde se carga el código de máquina (en hexadecimal) y los 8 valores sucesivos (también en hexadecimal) a partir de dicha posición de memoria.
A continuación vemos, separado por dos puntos, un número que representa al byte menos significativo de la suma de estos ocho valores antedichos. Esta suma es la llamada suma de control y sirve para verificar (en cierta forma) los valores ingresados.
Por supuesto, esto no garantiza que esté totalmente correcto, ya que puede darse que un error compense a otro, de a pares.
Nota importante:
El "Programa Cargador" se puede utilizar para ingresar cualquier programa en código de máquina y lo utilizaremos como base para las sucesivas publicaciones.

Breve explicación del programa Cargador:

Este programa tiene 2 modos:
M: Modificar memoria. permite cargar los datos del programa. Digitar los datos directamente en hexadecimal. Para retroceder, pulse <BS>. Para avanzar, barra espaciadora. Para salir de este modo, pulse <RETURN>
Ejemplo: M
d000
D000 7E -
Y aquí digite su número.
D: Dump de memoria. Permite verificar la posición de memoria indicada. Efectúa la suma de control e imprime los datos almacenados de a 8. Imprime 16 líneas por vez. Si desea continuar, pulse barra espaciadora. Para salir <RETURN>
Ejemplo: D
D000
D000 2A 76 F6 01 04 00 09 7E :22
D008 23 B7 28 15 D6 0B 38 F7 :27
Etcétera.
Q: finaliza el programa.

```
; Efectuar un CLEAR X,
&HD000 antes de ejecutarlo
;
; NOTA: s"lo funciona en
SCREEN 0
TITLECLOCK
;
; Inicializa HOOK H.TIMI

        ORG     0D000H
        LD      HL,0FD9FH
        LD      (HL),0CDH
        INC     HL
        LD      (HL),0CH
        INC     HL
        LD      (HL),0D0H
        RET
;
;
;
START:  PUSH    AF
        DI
```

```
        LD      A,(CON1)
        INC     A           ; Incrementa contador de 1/50 de seg
        CP      50
        JR      Z,L1        ; Si llego a 50 incrementar los seg.
        LD      (CON1),A    ; Si no retorna a BASIC
        POP     AF
        EI
        RET
L1:     LD      A,0         ; Contador de 1/50 de seg igual a 0.
        LD      (CON1),A
        LD      A,(CONS)
        INC     A           ; Incrementa segundos.
        CP      60
        JR      Z,L2        ; Si llego a 60 Incrementar minutos.
        LD      (CONS),A    ; Si no Imprimir.
        JR      IMP
L2:     LD      A,0
        LD      (CONS),A    ; Contador de segundos igual a 0.
        LD      A,(CONM)
        INC     A           ; Incrementa minutos.
        CP      60
        JR      Z,L3        ; Si llego a 60 incrementar horas.
        LD      (CONM),A    ; Si no Imprimir.
        JR      IMP
L3:     LD      A,0
        LD      (CONM),A    ; Contador de minutos igual a 0.
        LD      A,(CONH)
        INC     A           ; Incrementa hora.
        CP      24
        JR      Z,L4        ; Si llego a 24 colocar horas en 0.
        LD      (CONH),A    ; Si no imprimir
        JR      IMP
L4:     LD      A,0
        LD      (CONH),A    ; Contador de horas igual a 0.
IMP:    PUSH    HL
        PUSH    BC
        LD      A,(CONH)
        LD      HL,30
        CALL    PP          ; Imprime horas.
        INC     HL
        LD      A,3AH
        CALL    4DH         ; Imprime ":"
        INC     HL
        LD      A,(CONM)
        CALL    PP          ; Imprime minutos.
        INC     HL
        LD      A,3AH
        CALL    4DH         ; Imprime ":"
        INC     HL
        LD      A,(CONS)
        CALL    PP          ; Imprime segumdos.
        POP     BC
        POP     HL
        POP     AF
        EI
        RET     ; Retorna a BASIC.
```

```
PP:     CALL    DA
        LD      C,A
        LD      A,B
        ADD     A,30H
        CALL    4DH         ; Imprime parte alta del numero.
        LD      A,C
        ADD     A,30H
        INC     HL
        CALL    4DH         ; Imprime parte baja del numero.
        RET
DA:     LD      B,0         ; Calcula parte baja y alta del num.
D1:     INC     B           ;
        SUB     10          ; Divide por 10
        JR      NC,D1       ;
        DEC     B
        ADD     A,10
        RET
CON1:   DEFB    0
CONS:   DEFB    0
CONM:   DEFB    0
CONH:   DEFB    0
        END
```

```
D000 21 9F FD 36 CD 23 36 0C  :25   D038 3C 28 05 32 A0 D0 18 17  :3A   D070 3E 3A CD 4D 00 23 3A 9F  :8E
D008 23 36 D0 C9 F5 F3 3A 9E  :B2   D040 3E 00 32 A0 D0 3A A1 D0  :8B   D078 D0 CD 81 D0 C1 E1 F1 FB  :7C
D010 D0 3C FE 32 28 06 32 9E  :3A   D048 3C FE 18 28 05 32 A1 D0  :22   D080 C9 CD 93 D0 4F 78 C6 30  :B6
D018 D0 F1 FB C9 3E 00 32 9E  :93   D050 18 05 3E 00 32 A1 D0 E5  :E3   D088 CD 4D 00 79 C6 30 23 CD  :79
D020 D0 3A 9F D0 3C FE 3C 28  :17   D058 C5 3A A1 D0 21 1E 00 CD  :7C   D090 4D 00 C9 06 00 04 D6 0A  :00
D028 05 32 9F D0 18 29 3E 00  :25   D060 81 D0 23 3E 3A CD 4D 00  :06   D098 30 FB 05 C6 0A C9 00 00  :C9
D030 32 9F D0 3A A0 D0 3C FE  :85   D068 23 3A A0 D0 CD 81 D0 23  :0E   D0A0 00 00                    :00
```

Listado 2

```
100 SCREEN 0:CLEAR 200,&H87FF:Z$="0000"
110 ON ERROR GOTO 300
120 PRINT:PRINT "*";:GOSUB 260:PRINT A$
130 IF A$="M" THEN 150
135 IF A$="Q" THEN 310
140 IF A$="D" THEN 210 ELSE PRINT:GOTO 120
150 LINE INPUT A$:A=VAL("&H"+A$)
160 PRINT:GOSUB 280:V=PEEK(A):GOSUB 290:PRINT"-";
170 GOSUB 240:L=V*16:IF E=1 THEN 190 ELSE GOSUB 240:L=L+V:IF E=1 THEN 190 ELSE P
OKE A,L:A=A+1
180 GOTO 160
190 IF A$=CHR$(8) THEN A=A-1 ELSE IF A$=CHR$(32) THEN A=A+1 ELSE IF A$=CHR$(13)
THEN 120
200 GOTO 160
210 LINE INPUT A$:A=VAL("&H"+A$)
220 FOR L=0 TO 15:GOSUB 280:FOR M=0 TO 7:V=PEEK(A):S=S+V:GOSUB 290:A=A+1:NEXT:PR
INT":";:V=S:GOSUB 290:PRINT:NEXT
230 PRINT:GOSUB 260:IF A$<>" " THEN 120 ELSE 220
240 E=0:GOSUB 260:IF A$<CHR$(48) THEN E=1:RETURN ELSE IF A$>CHR$(70) THEN 240 EL
SE IF A$>CHR$(57) AND A$<CHR$(65) THEN 240
250 V=VAL("&H"+A$):PRINT A$;:RETURN
260 A$=INKEY$:IF A$="" THEN 260 ELSE IF A$>CHR$(96) AND A$<CHR$(123) THEN A$=CHR
$(ASC(A$)-32)
270 RETURN
280 A$=HEX$(A):PRINT LEFT$(Z$,4-LEN(A$))+A$+" ";:S=0:RETURN
290 A$=RIGHT$(HEX$(V),2):PRINT LEFT$(Z$,2-LEN(A$))+A$+" ";:RETURN
300 RESUME NEXT
310 ON ERROR GOTO 0:END
```




GUIA PRACTICA

DESENSAMBLANDO EL ASSEMBLER

PARTE III.

Grupo de carga de 16 bits

Hay que tener en cuenta que:
I) Ninguna de estas instrucciones afecta al indicador F.
II) dd es cualquiera de los registros BC, DE, HL o SP.
III) nn y (nn) son lo mismo que en el grupo de 8 bits.
Veamos:

LD dd,nn: copia el valor nn en el registro dd.
LD dd,(nn): copia en el registro de menor peso (LO) el contenido de la posición nn, y en el de mayor peso (HI) el contenido de la posición nn+1. Ejemplo: (65000.)=10. (65001)=32., luego de la ejecución de LD BC,(65000.) quedarán: B=32., C=10., y BC=8202. pues 8202.= 256.*32.+10.
LD (nn),dd: hace el camino inverso al anterior.
LD SP,HL: copia en SP el valor de HL.

Grupo aritmético y lógico de 8 bits

Tener en cuenta que r y n son los mismos que en el grupo de carga de 8 bits, y que s es cualquiera de r, n o (HL).
Algunos de estos son:
INC r: hace r=r+l. Afecta al indicador de estado, los bits Z, S y H cambian según el estado de esta operación.
INC (HL): hace (HL)=(HL)+l. Afecta al registro F según sea el resultado de esta operación, en los mismos bits que el anterior.
DEC r: Hace r=r−l. Afecta al flag de forma similar a los anteriores.
DEC (HL): Hace (HL)=(HL)−l. Afecta a F igual que los anteriores.
ADD s: hace A=A+s. Afecta a F al igual que los anteriores, pero también afecta al bit C (carry flag).
ADD (HL): Hace A=A+(HL). Afecta a F igual que el anterior.
SUB s: Hace A=A−s. F ídem anterior.
SUB (HL): Hace A=A−(HL). F ídem anterior.
CP s: Hace A−s pero no guarda el resultado en ningún registro, una vez hecha la comparación afecta a F como el anterior.
CP (HL): Hace A−(HL), ídem anterior.
OR s y OR (HL): Hace la operación lógica OR con el contenido del acumulador y guarda el resultado en este último. El bit C del flag es puesto en '0', el resto de los indicadores se mantienen con la misma variación que los anteriores.
Ejemplo: A=00101100 s ó (HL)= 10110001 luego de la operación OR s u OR (HL) el acumulador quedará A=10111101

Grupo aritmético de 16 bits

Recordemos que dd es igual que en el grupo de carga de 16 bits.
Veremos como siempre los de uso común.
ADD HL,dd: Hace HL=HL+dd. Sólo queda afectado según sea el resultado el indicador C del registro F.
INC dd: Hace dd=dd+l. No afecta al flag.
DEC dd: Hace dd=dd−l. No afecta a F.
Veremos ahora más detalladamente el grupo de salto (JUMP), cuyas instrucciones cumplen una función similar a la del GO TO.
La primera que veremos será JP nn, donde nn es una posición de memoria cualquiera entre 0. y 65535.
Esta hace saltar el control del programa a la posición nn, o sea es como un GO TO a la posición nn.
La operación simbólica es la de cargar el registro PC con nn, es buena costumbre releer temas anteriores al hacer referencia a éstos, pues se aclararán nuestros puntos oscuros.
Otra instrucción del mismo grupo un poco más elaborada es JP cc,nn donde cc es una de las condiciones que veremos en figura 1.
Todas estas condiciones se refieren al estado del bit correspondiente en el flag, que ha quedado como resultado de la operación anterior a JP. Veamos cómo trabaja en el programa de figura 2.
En primer lugar, llamaremos a L1 y a L2 "rótulos o etiquetas", que no son instrucciones sino que nos sirven como nombre de una determinada posición de memoria. Por ejemplo, L2 significa la posición que contiene al CM de la instrucción general LD HL,nn (posición 30002.). Entonces cuando queramos repetir el programa a partir de esta instrucción escribiremos JP L2, que en realidad es JP 30002.
Notamos que se usa dos veces la instrucción JP NZ,nn y no JP nn.
Vemos que los registros D y B son inicializados en 0., también se ve que, justo antes de JP NZ,L1, el contenido de B es disminuido en uno, como sabemos esta instrucción afecta al indicador de cero.
Pero lo que ocurre es que al hacer la resta 0−1 el registro B quedará con el número 255., con lo cual el resultado de esta operación por el momento no es cero. Entonces, al llegar el control a JP NZ,L1 "por **no** ser cero" el resultado de la operación anterior, el programa volverá a ejecutarse desde L1, donde está LD A,D.
Este ciclo se repetirá hasta que por fin al ejecutarse DEC B el contenido de éste sea cero, Jp NZ,L1 **no** se ejecutará y el control pasará a DEC D que afectará al indicador de cero y así dará paso al segundo bucle controlado por JP NZ,L2.
Será un buen entrenamiento pasar este programa a nuestro Spectrum o TS 2068, con el mismo método que utilizamos para los anteriores, es una pequeña locura de color.

Figura 1

cc	Condición
NZ	no cero
Z	cero
NC	no arrastre
C	arrastre
PO	paridad impar
PE	paridad par
P	signo positivo
M	signo negativo

El universo de instrucciones assembler del microprocesador Z80 se encuentra subdividido en grandes grupos. Continuamos viendo entonces, cuatro de estos grupos de comandos.

Figura 2

Posición de memoria	CM		Mnemónico
30000/1	22. 0.		LD D,0.
30002/3/4	33. 0. 89.	L2	LD HL,22784.
30005/6	6. 0.		LD B,0.
30007	122.	L1	LD A,D
30008	128.		ADD A,B
30009	119.		LD (HL),A
30010	35.		INC HL
30011	5.		DEC B
30012/13/14	194. 55. 117.		JP NZ,L1
30015	21.		DEC D
30016/17/18	194. 50. 117.		JP NZ,L2
30019	201.		RET

También será oportuno que juguemos un poco con los valores que controlan los bucles (el de D y B) y con el valor que se le asigna a HL, teniendo cuidado de no salirnos de la memoria destinada a atributos (22528.–23295) en el caso del Spectrum y del 2068.

Notemos lo siguiente: L1 representa a 30007.=256*(117)+(55), los números entre paréntesis son los que acompañan al CM 194. que corresponde a JP NZ,nn. Y L2 representa a 30002.=256*(117)+(50).

Entonces si quisiéramos reubicar el CM del programa anterior, sin cambio alguno, por ejemplo desde la posición 40000., obtendríamos un inevitable crash o bucle perpetuo, pues los valores que acompañan a los JP seguirán siendo 30002. y 30007.

Obviamente tendríamos que recalcular los valores de L1 y L2, caso que se complica a medida que aumentan los JP.

A este problema los creadores del Z80 le encontraron una "relativa" solución: Esta es el JR (Jump Relative) que puede trabajar con las mismas condiciones (cc) que el JP común, o sin ellas, pero que lleva a continuación un número (no mayor que 255.) que indica cuántos bytes más arriba o más abajo de dicha instrucción, se encuentra la posición de memoria a la que debe efectuarse el salto.

Esto hace que un programa que contenga este tipo de saltos, sea totalmente reubicable, con la restricción de **no** poder efectuar este salto más allá de 127. posiciones de memoria por encima o por debajo del byte que contiene al JR.

Antes de seguir con JR veamos qué es complemento a dos.

Este es un sistema de representación binaria que permite usar el primer bit (de izquierda a derecha) de un byte para indicar el signo del número que representan los bits siguientes.

En nuestro caso (8 bits) el número positivo de mayor valor representable será 127., y el menor (negativo) será −127.

Un número será entonces positivo si el primer bit es '0', y negativo si es '1'.

Para hallar el complemento de un número positivo cualquiera expresado en base binaria seguiremos este procedimiento:

Comenzando de derecha a izquierda dejaremos intactos los bits que contengan ceros hasta el primer uno que encontremos inclusive, luego cambiaremos el valor de los bits restantes por sus opuestos, esto es, donde halla un cero pondremos un uno y viceversa.

Por ejemplo, el complemento de 4.=00000100 es 11111100 que significa −4. en complemento, pero que en realidad es el número 252.

Veamos una tabla comparativa:

Número decimal	Representación en complemento
4	00000100
3	00000011
2	00000010
1	00000001
0	00000000
−1	11111111
−2	11111110
−3	11111101
−4	11111100

Volviendo a JR, este sistema de numeritos nos servirá para indicarle a esta instrucción dónde queremos que se efectúe el salto.

Supongamos que nuestra instrucción es JR 252. cuyo CM asociado es 24. 252., sabemos que 252. es el número decimal que representa en complemento a −4. Entonces JR 252. efectuará un salto cuatro posiciones más arriba de la instrucción siguiente a ésta, comenzando de cero. Veamos un dibujo que aclare esto:

	-4	← Instrucción que se ejecuta como resultado del salto.
	-3	
24.	-2	
252.	-1	
	0	← Instrucción que se ejecutaría a continuación.
	1	
	2	

Podríamos cambiar entonces en el programa anterior:
JP NZ,L1 cuyo CM es 194. 55. 117., por: JR NZ,L1 cuyo CM es 32. 249.
y
JP NZ,L2 donde su CM es 194. 50. 117., por JR NZ,L2 cuyo CM es 32. 241.

Como vemos, además de hacer reubicable nuestro programa esta instrucción nos hizo ahorrar dos bytes. Pronto le tomaremos la mano a esta ensalada de números y letras que nos dan la base para exprimir al máximo nuestras computadoras. Practiquemos sin miedo.

Nota: 249. es en complemento el número −7=11111001, y 241 equivale a −15 que en binario es 11110001. K64

Eduardo H. Mombello

LAS FUENTES DE ALIMENTACION

Si bien a primera vista no parecen ser importantes, deberemos tenerlas muy en cuenta si queremos incluir "hardware" adicional. Además suelen ser uno de los puntos más débiles de los equipos electrónicos.



A pesar de que ambas máquinas se alimentan con las mismas tensiones básicas: +5 y +12 volts, las fuentes de alimentación son completamente distintas por lo que las describiremos separadamente.

SPECTRUM:

La tensión de alimentación a la Spectrum llega desde la fuente externa (Fig. 1) que entrega 9 volts no regulados. De ella podemos extraer hasta 1,2 Amperes. Desde estos 9 volts se deben crear las siguientes tensiones: +5 volts con 5% de tolerancia para alimentar la lógica; +12 volts para los circuitos de video mientras que las memorias de video requieren también una alimentación negativa de −5 volts. Estas últimas tensiones deben estar dentro del 10% de tolerancia.

La alimentación de + 5 v.

En la figura número 2 podemos observar que la entrada desde la fuente externa que es cercana a los 9 V (ya que esta tensión no es regulada), se regula a +5 V por medio del regulador integrado 7805. Este regulador actúa como si fuese una resistencia variable manteniendo constante la tensión a su salida. Por ello la diferencia de tensiones entre la entrada y la salida, multiplicada por la corriente que circula es la potencia que se perderá en el disipador. Este último es una chapa de aluminio bastante grande que produce el calentamiento de la máquina.

La alimentación de + 12 V:

Dado que la alimentación primaria es de 9 V para producir los 12 V se debió recurrir a un oscilador para transformar la tensión continua en alterna, luego elevarla y nuevamente rectificarla regulándola a +12 V. A ello se debe el zumbido que se suele escuchar en la Spectrum. Observemos en la figura número 2 el oscilador formado por C43, R61, L1, TR4 y TR5. Cuando el sistema oscila la tensión inversa producida en la bobina L1 eleva la tensión en el colector de TR4 hasta un máximo de 13 volts. D15 se ocupa de la rectificación cargando a C44 quien se ocupará de mantener la carga durante el semiciclo restante. A través de la resistencia R59 se provee de la realimentación necesaria a TR5, el que por medio de la variación de la frecuencia mantendrá la tensión de 12 V contante.

La alimentación de − 5 V:

Ya que en el colector de TR4 tenemos una tensión alterna de 13 V C46 se carga a través de D11 a aproximadamente 12 V. Cuando la tensión del colector de TR4 pasa por cero, la placa negativa de C46 tiene − 12 V cargando a C47 a través de D12. El diodo Zener D16 se

Figura 1

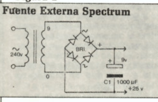
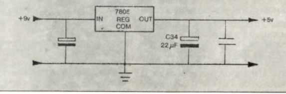


Figura 2



ocupa de mantener los −5 V constantes.

La alimentación de "− 12 V":

En el manual de la Spectrum se observa que el contacto número 23A está marcado con la inscripción −12 V lo que en realidad no es tan así: este contacto está conectado al colector de TR4 donde existe una tensión alterna de media onda de 12 V. Ocurre que la designación se ha realizado de esa manera ya que ese contacto se utilizará en la Interfase I para la creación de los −12 V necesarios para la salida RS-232. Dentro de la Interfase I se encuentra implementado un circuito similar al de la fuente de −5 V.

Límites de corriente:

En la Spectrum de 48 K las fuentes de alimentación están bastante exigidas por lo que si queremos agregar algún circuito será conveniente la utilización de una fuente externa. En lo que respecta a la fuente externa de la Spectrum, ésta sí tiene reserva; se pueden obtener 0,5 Amperes adicionales de ella, por lo que se puede agregar un 7805 con un par de capacitores de 25 microfaradios a la entrada y salida del mismo y de esta forma alimentamos nuestro "hardware".

TS 2068:

La fuente externa de la TS 2068 es similar a la de la figura 1 aunque su tensión de entrada es de 117 V CA. Esta fuente entrega desde 14 a 25 V con un consumo de 1 Ampere ante variaciones de la tensión de alimentación de 105 a 130 V CA. La tensión de aislación del transformador es superior a los 1500 V. Desde esta tensión que normalmente ronda los 20 V se obtienen dos voltajes: + 12 V para los circuitos de video y + 5 V para la lógica restante.

La alimentación de + 5 V:

En esta parte del circuito de la TS 2068 es donde se utiliza tecnología de punta en fuentes de alimentación: un regulador conmutado tipo uA 78S40.
En este tipo de fuente un conmutador estático capaz de conectar y desconectar, transforma la tensión continua de entrada en continua pulsante rectangular. La regulación se realiza variando la relación entre el tiempo de conducción del conmutador y el período completo. Estas fuentes de alimentación poseen un rendimiento muy elevado y a diferencia de los reguladores serie no hay energía calorífica a disipar, lo que es fácilmente apreciable ya que la máquina está siempre "fresca".
En la figura número 4 se puede observar el diagrama funcional del circuito integrado uA 78S40, el que contiene un oscilador, comparador de tensiones que actúa variando el

Figura 3

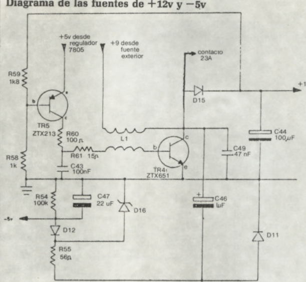
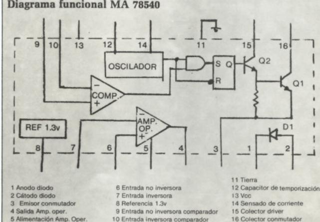


Figura 4



tiempo de conducción y una etapa de salida que comanda el transistor de potencia. Este integrado también posee una fuente de referencia de 1,3 V y un amplificador operacional que muy inteligentemente se utiliza para excitar al parlante. Existe, además, una protección contra sobrecorriente (pata 14) que frenará el oscilador ante sobrecargas.

En la figura número 5, se muestra el círculo completo de la fuente de alimentación de la TS 2068.

La alimentación de 12 V:

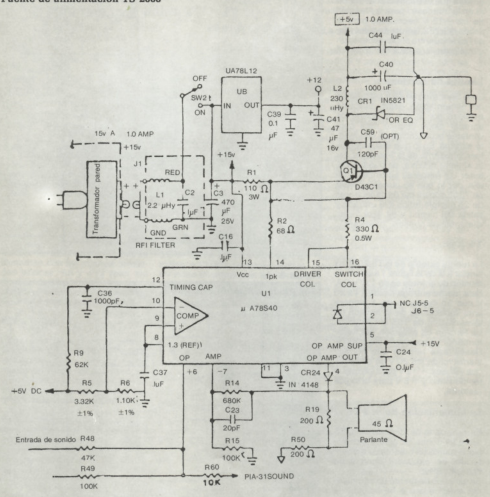
Según observamos en la figura número 5 esta se obtiene de un regulador integrado serie tipo UA78L12 capaz de proporcionar hasta 100 mA.

Límite de corriente:

Aquí valen las mismas consideraciones que para la Spectrum con respecto a la tensión de + 5 V. Dado que el consumo de la máquina desde la fuente externa es de 220 mA, podremos extraer 800 mA adicionales para otros usos.

G.E.

Figura 5



SOFTWARE MEDICO

El doctor Eduardo Yañez nos explicó la utilización que le dio a micro computadoras hogareñas en un servicio de terapia intensiva


Dr. Eduardo Yáñez - Hospital Bartolomé Churruca

Me interesé por la computación porque escuché que había gente que estaba trabajando en esto y mi jefe tenía intención de aplicarla a la medicina. Pero no tenía idea de lo que era una computadora. Sabía que era un aparatito con teclitas, y empezando a leer revistas y libros, entré en el mundo de la computación antes de tener la máquina. Primero empezó mi jefe y a los dos meses me compré la computadora yo; inmediatamente empezamos a estudiar y aprender a programar. No nos interesaba la computadora como algo prehecho, con software establecido. Porque muchas cosas que nosotros necesitamos en medicina, no existen en programas, no hay software.

¿Qué le podíamos sacar a la computadora así como venía? Comprar un fichero-archivo y una base de datos y punto. Los programas de estadística que había, que a nosotros nos interesaban muchísimo por los trabajos científicos, eran muy pobres. Había muy pocos test estadísticos, muy limitados.

Entonces nos propusimos empezar a trabajar sobre un programa de estadística. Como éramos autodidactas, nos costó un trabajo terrible al principio. Fuimos aprendiendo y al final, terminé mi programa. Mi jefe, que tenía una Spectrum se quedó y no lo pudo terminar.

Con las limitaciones que me daban los 16K de memoria pude hacer un programa que tiene un 60% de unidad estadística, que desde ya era muchísimo más completo que cualquier programa que ofrecen los comercios. Pero no me permitió hacer todo lo que deseaba.

Cuando terminé de hacer el programa me decidí a comprar la Commodore. Entonces perdí todo, porque el programa que tenía grabado en cassette ya no era compatible.

Reiniciamos el trabajo, pero con mucho más conocimiento, lo que nos permitió que los programas fueran más versátiles. Si Dios quiere vamos a tener terminado el programa de aquí a un par de meses y con la estadística completa, incluyendo gráficos. Con una versatilidad mayor en cuanto a la entrada de datos para que se pueda cambiar, corregir y trabajar con un número menor de datos que los ingresados.

Si yo ingreso 20 datos de una muestra y 20 de otras, pero después quiero sacar datos que no interesan o pueden falsear la muestra, entonces los puedo retirar. Todo eso lo podemos hacer porque tenemos más capacidad de memoria. La falta de tiempo no lo hace fácil. Yo le puedo dedicar poco tiempo, dos horas tres veces por semana. Pero lo vamos logrando y bastante bien. En esto tiene mucho que ver las revistas que hay aquí en la Argentina, y en especial **K-64**.

Leer los programas de juegos, en **K-64**, aún siendo de otras computadoras, a mí me enseñó muchísimo. Porque el software no lo usábamos para jugar; los desmenuzábamos y aprendíamos. Hemos llegado al punto de escribir los programas sin las computadoras. En el rato que quedaba luego de terminar la labor asistencial, nos poníamos a escribir el programa; después generalmente entraba en la computadora sin problemas.

Nosotros veníamos al servicio con nuestras computadoras. Yo traía la TK, mi compañero traía la Spectrum, y como lo que hacía no usaba un lenguaje de máquina, lo que escribíamos era compatible con las dos máquinas, excepto pequeños detalles.

Proyectos

Ahora somos mucho más ambiciosos; no queremos terminar sólo con un programa estadístico. Nos va a costar, porque es muy complejo, pero nuestra idea es tener una gran base de datos en el Servicio, en el que podemos incluir todos los parámetros hemodinámicos, medio interno de los pacientes, etc.

Después podremos extraer lo que necesitamos, y hacer la estadística. El asunto es engancharlo con el programa.

¿El programa de estadística para qué nos sirve? Para que cada tanto, cuando vamos a hacer un trabajo científico, podamos demostrar "tal cosa". Pero nos sirve solamente para un grupo de enfermos. Nosotros queremos tener una gran base de datos donde estén presentes todos los parámetros. Pero es mucho más lo que deseamos hacer. Como alguien decía, en computación "su límite es su imaginación".

UNA SUBRUTINA DE PLOTEO

Ing. Jorge Valle



Es bastante corta, rápida y nos permite realizar gráficos de una manera sencilla.

Si bien es cierto que la TI-99 no tiene buenas posibilidades de graficación, al menos desde el BASIC, es posible realizar algunas subrutinas que, aunque no tengan la rapidez que podría brindar una subrutina de ASSEMBLER, nos permite realizar gráficos de una manera sencilla.

Hay varias maneras de diseñar un programa para graficar puntos. Esta subrutina es bastante corta y algo más rápida que otras que conozco. De todos modos tal vez algún lector con ingenio le pueda quitar varias instrucciones y, tal vez, hacerla un poco más rápida.

Un poco de geometría

La pantalla es un mundo de 256 puntos de ancho por 192 de alto. Cada uno de ellos puede ser prendido por separado. Debido a que este computador no puede manejar cada pixel en forma independiente, sino que modifica todo un carácter de 64 pixels, hay que hacer varios cambios de coordenadas para llevar nuestro punto (X, Y) de las coordenadas generales de la pantalla, a las coordenadas del carácter donde se encuentra. Luego hay que calcular las coordenadas locales dentro del carácter.

El orígen de coordenadas de la pantalla está en el extremo superior izquierdo. En nuestro caso resulta más cómodo que el orígen esté en el centro de la pantalla, y que la coordenada horizontal se llame X y la vertical Y. Vamos a llamar a estas nuevas coordenadas H y V:

H=128+X

V=96−Y

A partir de esas coordenadas generales vamos a determinar las coordenadas del carácter donde debe colocarse el punto:

VC=INT (V/8)+1

HC=INT (H/8)+1

Una vez determinada la posición del carácter necesitamos calcular las coordenadas locales dentro del carácter donde se encuentra

```
10000 V=96-Y::H=128+X
10010 VC=INT(V/8)+1::HC=INT(H/8)+1
10020 HL=H-(HC-1)*8::VL=V-(VC-1)*8
10030 CALL GCHAR(VC,HC,COD)
10040 IF COD=32 THEN NC=NC+1::COD=NC::CALL
CHAR(COD,A$)
10050 CALL CHARPAT(COD,X$)
10060 I=2*VL+1+INT(HL/4)
10070 H$=SEG$(X$,I,1)
10080 HP=HL-INT(HL/4)*4
10090 P=INT(2^(3-HP))
10100 IF ASC(H$)>57 THEN K=((ASC(H$)-7)
OR P)+7::GOTO 10140
10110 K=ASC(H$) OR P
10120 IF K>57 THEN K=K+7
10130 Z$=SEG$(X$,1,I-1)&CHR$(K)&SEG$
(X$,I+1,16-I)
10140 CALL CHAR(COD,Z$)
10150 CALL HCHAR(VC,HC,COD)
10160 RETURN
```

El siguiente programa realiza una curva muy interesante:

```
100 CALL CLEAR
110 NC=32
120 FOR TT=0 TO 360
130 T=TT*PI/180
140 X=INT(40*COS(5*T))
150 Y=INT(40*SIN(7*T))
160 GOSUB 10000
170 NEXT TT
180 GOTO 180
```

nuestro punto. Como sabemos cada carácter está definido por 16 números hexadecimales, donde cada uno de ellos representa cuatro pixels. Necesitamos conocer cuál de los 64 pixels es el nuestro y cuál hexadecimal le corresponde:

HL=H - (HC - 1) * 8
VL=V - (VC - 1) * 8
I=2 * VL+INT (HL/4)+1

HL y VL son las coordenadas locales e I es la posición del hexadecimal.

Ahora podemos determinar qué número de carácter tiene asignada esa posición de la pantalla y luego qué dibujo tiene ese carácter.

CALL GCHAR (VC, HC, COD)
CALL CHARPAT (COD, X$)

En COD tenemos el código ASCII y en X$ el string que define su dibujo. De los 16 hexadecimales que lo definen, el de la posición I es el que tenemos que reemplazar.

Como cada hexadecimal representa cuatro puntos, tenemos que reemplazar el nuestro y dejar el resto como está. De este modo nuestras curvas podrán cortarse a sí mismas sin alterar el dibujo.

H$=SEG$ (X$, I, 1)
HP=HL - INT (HL/4) * 4
P=INT (2^(3-HP))

En H$ tenemos el hexadecimal, en HP la posición del bit que hay que reemplazar (de 0 a 3), y en P calculamos el número binario correspondiente a un 1 en la posición de nuestro punto y 0 en las demás. Una operación lógica OR entre este binario y el hexadecimal determina la nueva combinación de ceros y unos.

Haciendo algunas operaciones sencillas con el código ASCII del hexadecimal lo llevamos a un valor en el que la operación OR nos de el resultado buscado.

Sólo resta armar nuevamente el string que define el carácter, reemplazando el hexadecimal calculado, asignarlo al código ASCII correspondiente y colocarlo en pantalla.

Como la cantidad de caracteres disponibles está limitada a 110 (de 32 a 143), los dibujos no podrán ocupar más que esa cantidad de caracteres.

Para evitar una pregunta dentro de la subrutina, es necesario definir NC=32. Esto se debe colocar antes del llamado a la subrutina en un lugar donde se pase una sola vez.

Las curvas de Lissajou

Una aplicación simpática y al mismo tiempo interesante desde el punto de vista conceptual la brindan las curvas de Lissajou. Estas se obtienen al combinar dos ondas senoidales cruzadas a 90 grados. Las ecuaciones que las rigen son:

X=RX * COS (WX * T+AX)
Y=RY * SEN (WY * T+AY)

El parámetro T es un valor angular que varía entre 0 y 360 grados. Si RX=RY y WX=WY se obtiene una circunferencia. Si se hace que RX sea distinto de RY se obtendrá una elipse. De la misma manera, al tomar valores distintos de WX y WY se obtiene toda una familia de curvas que sin duda poseen un encanto muy particular.

AX	AY	WX	WY	
0	0	1	1	circunferencia
45	0	1	2	parábola
45	0	1	1	elipse
0	0	1	2	mono

K64

MODIFICACION DE LOS PUNTEROS



El intérprete basic de la C-64 utiliza una serie de direcciones de memoria denominadas "pointers" (punteros) las cuales apuntan, a su vez, a otras direcciones, donde se hallan determinadas rutinas que son utilizadas por él. Como ejemplo podemos citar las direcciones $308-$309 cuyos contenidos señalan a la dirección donde se encuentra la rutina que ejecuta un comando Basic. Como ese puntero se halla dentro de la memoria RAM, podemos modificarlo para agregar comandos propios. La rutina que nombramos anteriormente toma un byte llamado "token" el cual representa un determinado comando. Luego lo utiliza para acceder a la direcciones donde él se encuentra para, así, poder ejecutarlo. Cada uno de estos tiene su "token" respectivo. Por ejemplo, el token del comando PRINT es $99 (recordemos que el símbolo "$" significa que se trata de un número hexadecimal). La ventaja de tener codificados los comandos es que, en el momento de ejecutarse, se evita el hecho de comparar caracter por caracter obteniéndose, así, mayor velocidad de ejecución. Volviendo al caso de PRINT, y si optamos por el método ineficiente, debemos ir comparando la P,R,I,N y T para luego ejecutarlo. Imaginen el tiempo que tardaría la ejecución de cada comando. De la otra manera, sólo debemos comparar con $99 (su token); una sola comparación. ¿Cómo codifica el intérprete un determinado comando a su respectivo token? La respuesta es utilizando un puntero. En este caso, éste se halla en la dirección $304-$305 y señala a la rutina que realiza lo último descripto. Este método tiene aún otra ventaja: ahorro de memoria. Para comprender esto, apartémonos por un momento del tema aquí tratado, visualizando cómo se almacena una línea de programa en la memoria del C-64. Los primeros dos bytes representan la dirección de la próxima línea en formato byte bajo-byte alto. El tercer y cuarto byte representan el número de línea (también byte bajo-alto). El quinto byte corresponde al token de un determinado comando. En caso de que sea una asignación del tipo X=, en esta posición se pone el código ASCII de la variable en cuestión. Todo el texto que siga de aquí en más se codifica en su correspondiente ASCII. Como fin de línea se pone 00. Por ejemplo, si el programa que tenemos en memoria es:

10 PRINT"HOLA"

se almacenará como:

0801 0C byte bajo próxima línea
0802 08 byte alto próxima línea
0803 00 número de línea byte bajo-alto
0804 0A (línea 10)
0805 99 token PRINT
0806 22 comillas
0807 48 ASCII de H
0808 4F ASCII de O
0809 4C ASCII de L
080A 41 ASCII de A
080A 22 comillas
080B 00 fin de línea
080C 00 fin de programa, también se indica con 00.

Supongan que, en lugar de $99 tendrían que ir los caracteres que forman el comando. Necesitaríamos correr cinco bytes hacia abajo, disminuyendo así la memoria libre. Regresemos a los punteros del C-64. Los más importantes son:

Puntero	Dirección	Descripción
$300-$301	$E38B	Arranque en caliente del Basic y entrada de errores.
$302-$303	$A483	Espera de ingreso de nueva línea.
$304-$305	$A57C	Convierte una línea a Token.
$306-$307	$A71A	Convierte Token en el comando.
$308-$309	$A7E4	Ejecuta comando Basic.
$30A-$30B	$AE86	Evalúa expresión Basic.

Nosotros vamos a analizar uno de ellos: el puntero $300-$301. El señala a la dirección $E38B. Aquí se salta cada vez que se finaliza un programa o que se encuentra un error (sintaxis, etcétera). En este caso, el intérprete carga en el registro X el número de error y salta

La siguiente nota nos introduce en los punteros de la C-64. Explica en qué consisten y cómo modificarlos para crear nuevos comandos y traducir los errores al castellano.

a la dirección $A437 donde se halla el salto indirecto a la $E38B (es decir realiza JMP ($0300)).
Cada error, como antes dijimos, está codificado con un número. Ellos son:

Número (en hexa) **Descripción**

1 TOO MANY FILES
2 FILE OPEN
3 FILE NOT OPEN
4 FILE NOT FOUND
5 DEVICE NOT PRESENT
6 NOT INPUT FILE
7 NOT OUTPUT FILE
8 MISSING FILENAME
9 ILLEGAL DEVICE NUMBER
A NEXT WITHOUT GOSUB
B SYNTAX ERROR
C RETURN WITHOUT GOSUB
D OUT OF DATA
E ILLEGAL QUANTITY
F OVERFLOW ERROR
10 OUT OF MEMORY
11 UNDEF'D STATEMENT
12 BAD SUBSCRIPT
13 REDIM'D ARRAY
14 DIVISION BY ZERO
15 ILLEGAL DIRECT
16 TYPE MISMATCH
17 STRING TOO LONG
18 FILE DATA
19 FORMULA TOO COMPLEX
1A CAN'T CONTINUE
1B UNDEF'D FUNCTION
1C VERIFY
1D LOAD

Los mensajes de error del 1 al 9 se refieren a errores de entrada/salida y son trasmitidos por el sistema operativo. Los restantes proceden del intérprete basic. ¿Será posible, pues, imprimir los mensajes en castellano? La respuesta está directamente relacionada con la posibilidad de modificar el puntero respectivo ($300-$301). Y, al estar en RAM, se puede hacer. Entonces, trataremos de modificar por ahora, el mensaje de SINTAX ERROR. En lugar de él imprimiremos ERROR. No debemos olvidar que el primero de ellos se imprime tanto en modo directo como en modo ejecución. Es decir, que si realizamos en modo directo:

PRONT 12*2

el intérprete nos responderá con SINTAX ERROR, mientras que

10 PRONT 12*2

responderá con SINTAX ERROR IN 10. Nosotros, para este caso, pondremos ERROR EN 10. Necesitaremos, además, trabajar con una rutina que imprime en pantalla un número de línea. Antes de llamarla debemos almacenar el byte bajo en el registro X y el byte alto en el acumulador. La rutina se encuentra en la dirección $BDCD. Por ejemplo, para imprimir el número correspondiente a $000A (10) debemos hacer:

```
LDA #$0A
LDX #$00
JSR $BDCD
RTS
```

También trabajaremos con las direcciones $0039-$003A. Aquí se almacena el número de línea que actualmente se está ejecutando. Si el sistema se encuentra en modo directo, en la dirección $003A se coloca el valor de $FF. Si estamos en modo ejecución se pondrá el número de línea siempre en formato byte bajo-alto, (es decir $0039 tendrá la parte baja, mientras que, $003A la parte alta).

Otra de las subrutinas necesarias es la que se encuentra en la dirección $FFD2. La función de ella es imprimir en pantalla en caracter almacenado en el acumulador.

Debemos hacer entonces:

1) Modificar el puntero a la dirección donde se encontrará el programa respectivo.
2) Efectuar la traducción:
2.1) Ver si el registro X contiene el código $0B (sintax error). Si es, pasamos al punto 2.2. Si no, es otro error y lo derivamos a la rutina original.
2.2) Imprimimos el mensaje ERROR.
2.3) Determinamos si estamos en modo directo. Si es así, cargamos el registro X con $80 y saltamos a la dirección $A437. Esto ocasionará que se imprima el mensaje READY. Si no estamos en este modo pasamos al punto 2.4
2.4) Imprimimos EN
2.5) Tomamos el número de línea actual de las direcciones $39-$3A.
2.6) La imprimimos.
2.7) Saltamos a la dirección que provoca el mensaje READY.

El programa correspondiente al punto 1 lo ubicaremos a partir de la dirección $C000:

```
C000 A9 00    LDA #$00
C002 A2 C1    LDX #$C1
C004 8D 00 03 STA $0300
C007 8E 01 03 STX $0301
C00A 00       BRK
```

Al ejecutar este programa cambiaremos la dirección que señala el puntero $300-$301. Ahora, apunta a la dirección $C100, donde se encuentra el programa correspondiente al punto 2, cuyo listado es el que sigue a continuación:

```
C100 E0 0B    CPX #$0B      ;si es SINTAXS ERROR
C102 F0 03    BEQ $C107     ;saltamos a la direccion $C107
C104 4C 8B E3 JMP $E38B     ;si no, ejecutamos normalmente
C107 A2 00    LDX #$00      ;reseteamos al registro X
C109 BD 00 C2 LDA $C200,X   ;cargamos uno de los caracteres de "ERROR"
C10C C9 00    CMP #$00      ;el 0 indica fin de texto
C10E D0 25    BNE $C135     ;si no finalizo vamos a la $C135
C110 A5 3A    LDA $3A       ;si finalizo, vemos si estamos en modo directo
C112 C9 FF    CMP #$FF      ;comparando con $FF
C114 D0 06    BNE $C11C     ;si es modo ejecucion vamos a la $C11C
C116 A2 80    LDX #$80      ;si es modo directo ocacionamos el READY
C118 4C 37 A4 JMP $A437     ;luego de la impresion de "ERROR"
C11B EA       NOP           ;no operamos
C11C A9 45    LDA #$45      ;cargamos el acumulador con ASCII de "E"
C11E 20 D2 FF JSR $FFD2     ;lo sacamos por pantalla
C121 A9 4E    LDA #$4E      ;cargamos el acumulador con ASCII de "N"
C123 20 D2 FF JSR $FFD2     ;lo sacamos por pantalla
C126 A9 20    LDA #$20      ;cargamos acumulador con el ASCII del carac. espacio
C128 20 D2 FF JSR $FFD2     ;lo sacamos por pantalla
C12B A5 3A    LDA $3A       ;cargamos el acumulador con el numero de
C12D A6 39    LDX $39       ;linea donde se produjo el error
C12F 20 CD BD JSR $BDCD     ;lo imprimimos
C132 4C 16 C1 JMP $C116     ;vamos a la $C116; ocacionamos el READY
C135 20 D2 FF JSR $FFD2     ;viene de la $C10E; imprimimos el caracter actual
C138 E8       INX           ;incrementamos para tomar el proximo caracter
C139 4C 09 C1 JMP $C109     ;regresamos a la $C109
C13C 00       BRK
```

PELOTA AL BLANCO

COMP.: DREAN COMMODORE 64
CLAS.: ENT



Debemos tirar la pelota al blanco, ayudándonos con las paletas que aparecen en pantalla. Para jugar tendremos que utilizar las teclas M y N. Si deseamos eliminar una paleta deberemos pulsar la barra espaciadora cuando la pelota toca esa paleta.

```
1 PG$="PELOTA AL BLANCO":AU$="S.LARSEN":
A2$="F.D.PINEIRO Y LEONE":BG$="RETURN"
3 REM AS OF 25JUL84 1984
4 REM COPYRIGHT (C) 1984 THE CODE WORKS
5 REM BOX 6905,SANTA BARBARA, CA 93160
6 REM MOD.& TRADUCED BY F.D.PINEIRO Y LE
ONE
7 REM AS OF 005OCT85
10 REM ***LISTA DE VARIABLES***
11 REM A=LOCALIZACION PELOTA.*BC=VALOR P
OKE P/B.ESPACIAD.*BL=VALOR POKE P/PELOTA

12 REM D%=DIRECC.D/PELOTA.*DE=RULO DE PA
USA VARIABLE.*M()=DURACION DEL JUEGO.
13 REM MAX=MAXIMO NRO.DE IMPACTOS LOGRAD
OS.*MS()=ACUMULADOR DE TIEMPO.*SK=NIVEL.

14 REM NT=NRO.D/IMPACTOS JUEGO EN CURSO.
*T$()=SEGUNDOS SUMADOS A TI$(DEP.NIVEL).

15 REM VALOR POKE DE BLANCOS EN IMAGEN.*
TC=CONSTANTE DE MEMORIA COLOR.
20 REM ***MODO DE TRABAJO D/PROGRAMA***
21 REM 100-120:SELECCIONA NIVEL.*130-180
:ENCIENDE 'SID'.*190-240:DET.COLORES.
22 REM 250-280:DIBUJA BORDES ALR.IMAGEN.
*290-310:DET.POS.PELOTA Y SU TRAYECTORIA

23 REM 320-360:RULO D/PROG.PPAL.*370+380
:MUEVE CHQ.LOCALIZ.PELOTA P/INSP.NUEVA.
24 REM 390-460:RECOJE INF.TECLADO Y ACTU
A EN CONSEC.*470-500:COPUTA NVA.LOC.PEL.

25 REM 510-540:DIBUJA PELOTA Y CONT.RULO
 D/PROG.PPAL.*550-590:SELEC.SONIDO.CPTE
26 REM 600-620:PELOTA HACE BLANCO.*630-7
00:PELOTA TOCA PALETA.*710-790:FINALIZA.

27 REM 800-850:DETERMINA POSICION BLANCO
.*860-900:SONIDO DE IMPACTO EN BLANCO.
28 REM 920-1000:SONIDO FIN TIEMPO.*1010-
1090:SELEC.NIVEL D/JGO.*1100-1130:TIEMPO
29 REM 1150-1320:ENCIENDE SET D/CARACTER
ES.RUTINA SYS COPIA SET STANDARD.
90 GOTO 62000
100 HG=24:RS=160:DEF FN R(R)=RND(1)*R:T0
=102:BL=81:BC=32:K4=78:K6=77
110 GOSUB 1150:GOSUB 1100
120 GOSUB 1010
130 FOR Z=SID TO SID+24:POKEZ,0:NEXT
140 POKE SID+0,71:POKE SID+1,5
150 POKE SID+12,85:POKE SID+13,85
160 POKE SID+5,6:POKE SID+6,6
170 POKE SID+9,0:POKE SID+10,5
180 POKE SID+24,15
190 TC=CM-CRT
200 NT=0:NG=NG+1
210 PRINT"□■"
220 POKE VIC+32,14:POKE VIC+33,1
230 POKE VIC+24,31
240 GOSUB 800
250 FOR I=1 TO WD:Z=(CRT+I-1):POKE Z,RS:
POKE TC+Z,6:Q=(CRT+1-1+(WD*HG))
260 POKE Q,RS:POKE TC+Q,6:NEXT
270 FOR I=1 TO HG-1:Z=(CRT+(WD*I)):POKE
Z,RS:POKE TC+Z,6:Q=(CRT+(WD*(I+1))-1)
280 POKE Q,RS:POKE TC+Q,6:NEXT
290 D%=FN R(4):IF D%=0 THEN D%=4
300 A=(CRT+WD*HG/2)+WD/2:IF A=T THEN POK
E (T+41),T0
310 POKE A,BL:TI$="0000"+T$(SK)
320 PRINT"■"TAB(3)"■ BLANCO      "NT
330 TT$=TI$:PRINT"■"TAB(WD-14)"■ TIEMPO:
";
340 PRINT TAB(26)M(SK)-VAL(MID$(TT$,4,1)
)"■:";
350 PRINT RIGHT$("0"+MID$(STR$(59-VAL(MI
D$(TT$,5))),2),2)
360 IF VAL(TT$)>MS(SK) THEN 920
370 A0=A:ON D% GOSUB 470,480,490,500
380 C=PEEK(A):IF C>99 THEN 510
390 GET I$:IF I$=""THEN 510
400 IF I$="Q"THEN 710
410 IF I$="N"THEN POKE A,K4:GOTO 450
420 IF I$=" "THEN POKE A,BC
430 IF I$="M"THEN POKE A,K6:GOTO 450
440 GOTO 460
450 POKE TC+A,0
460 C=PEEK(A):GOTO 510
470 A=A+1:RETURN
480 A=A+WD:RETURN
490 A=A-1:RETURN
500 A=A-WD:RETURN
510 IF C<>BC THEN 550
520 POKE A,BL:POKE TC+A,8
530 IF PEEK(A0)=BL THEN POKE A0,BC
540 GOTO 330
550 POKE SID,INT(RND(1)*60)
560 IF C<>T0 THEN POKE SID+4,129
570 POKE SID+4,128:IF C<128 THEN 600
580 GC=FRE(0):D%=D%+2:IF D%>4 THEN D%=D%
-4
590 POKE A0,BC:A=A0:GOTO 330
600 IF C<>T0 THEN 630
610 POKE TC+A,8:NT=NT+1:RT=RT+1:POKE T,B
L:POKE A0,BC
620 TT$=TI$:GOSUB 860:GOSUB 800:TI$=TT$:
GOTO 320
630 IF C=K4 THEN 660
640 IF(D%=1)OR(D%=3)THEN D%=D%+1:GOTO 70
0
650 IF(D%=2)OR(D%=4)THEN D%=D%-1:GOTO 70
0
660 IF(D%=1)OR(D%=3)THEN D%=D%-1:GOTO 68
0
670 IF(D%=2)OR(D%=4)THEN D%=D%+1
680 IF D%=0 THEN D%=4
690 IF D%=5 THEN D%=1
700 POKE A0,BC:GOTO 330
710 POKE VIC+24,21
720 PRINT"□■■NUMERO DE BLANCOS ESTE JGO.
:"NT
730 IF NT>MAX THEN MAX=NT
740 PRINT"■■MAYOR PUNTAJE:"MAX
750 GET Z$:IF Z$<>""THEN 750
760 PRINT"■■■■QUIERE JUGAR OTRO (S/N)?";
:GOSUB 60000:IF IN$="Q"THEN 60600
770 IF IN$=""OR IN$="S"THEN GC=FRE(0):GO
TO 200
780 POKE VIC+24,21
790 GOTO 60600
800 T=INT(FN R(WD*HG)+CRT)
810 IF(T<(CRT+WD))OR(T>(CRT+WD*(HG-1)))T
HEN800
820 FOR I=1 TO HG-1:Z=(CRT+(WD*I)):Q=(CR
T+(WD*(I+1))-1)
830 IF (Z=T)OR(Q=T)THEN 800
840 NEXT:POKE T,T0:POKE TC+T,6
850 RETURN
860 POKE SID+7,0
870 POKE SID+11,65
880 FOR Z=0 TO 70:POKE SID+8,Z:NEXT Z
890 POKE SID+11,0
900 RETURN
920 POKE SID+14,244:POKE SID+15,15
930 POKE SID+19,7:POKE SID+20,7
```




```
940 FOR Z=1 TO 17
950 POKE SID+18,33
960 FOR DE=1 TO 10:NEXT DE
970 POKE SID+18,32
980 FOR DE=1 TO 12:NEXT DE
990 NEXT Z
1000 GOTO 710
1010 POKE VIC+32,14:POKE VIC+33,1:PRINT"
"
1012 PRINT"PARA JUGAR UTILICE LAS TECLAS
 'M' Y 'N'."
1014 PRINT"SI DESEA ELIMINAR UNA PALETA,
PULSE BARRA"
1016 PRINT"ESPACIADORA CUANDO LA PELOTA
LA TOCA."
1018 PRINT"TOMESELO CON CALMA!!!"
1020 PRINT"1> FACIL"
1030 PRINT"2> MEDIANO"
1040 PRINT"3> DIFICIL"
1050 PRINT:PRINT"NIVEL DESEADO? ";:GOS
UB 60000:IF IN$=""THEN IN$="2"
1060 T=VAL(IN$)
1070 IF IN$="0"THEN 60600
1080 IF T<1 OR T>3 THEN PRINT"TTT":GOTO
1050
1090 SK=T:RETURN
1100 T$(1)="30":MS(1)=158:M(1)=1
1110 T$(2)="00":MS(2)=58:M(2)=0
1120 T$(3)="30":MS(3)=58:M(3)=0
1130 RETURN
1150 PRINT"EN MARCHA....."
1160 B=49152:I=0
1170 READ V:IF V=-1 THEN 1190
1180 POKE B+I,V:I=I+1:GOTO 1170
1190 SYS 49152
1200 B=14336
1210 READ P:IF P=-1 THEN 1250
1220 B2=B+P*8
1230 FOR Z=0 TO 7:READ V:POKE B2+Z,V:NEX
T Z
1240 GOTO 1210
1250 RETURN
1260 DATA 120,165,1,41,251,133,1,169,0,1
33,251,133,253,169,56
1270 DATA 133,252,169,208,133,254,162,8,
160,0,177,253,145,251
1280 DATA 200,208,249,230,252,230,254,20
2,208,240,165,1,9,4
1290 DATA 133,1,88,96,-1
1300 DATA 81,0,124,254,254,254,254,124,0

1310 DATA 102,24,24,24,231,231,24,24,24
1320 DATA -1
60000 IN$="":ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$(" ",ZC,
1);"";:ZC=3-ZC:ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF(Z AND 127
)<32 THEN PRINT" ";:GOTO 60110
60090 IF ZL>=QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>0 THEN IN$=LEFT$(IN
$,ZL-1):PRINT"";:GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>0)):
FOR Z=1 TO ZL:PRINT Z$;:NEXT:GOTO 60000
60140 GOTO 60010
60200 ZJ=TI+30:ZT=TI:ZS=2:PRINT LEFT$(JC
$,1);PR$;"T"
60210 IF(PEEK(JS) AND 16)=0 THEN ZS=2:GO
SUB 60280:PRINT:RETURN
60215 GOSUB 60500
60220 IF TI>=ZT THEN GOSUB 60280:ZT=TI+1
5:ZS=3-ZS
60230 Z=PEEK(JS)AND 12:IF Z=12 THEN ZJ=0
:GOTO 60210
60240 IF TI<ZJ THEN 60210
60250 IF Z=4 AND IN<JM THEN IN=IN+1:GOTO
 60200
60260 IF Z=8 AND IN>1 THEN IN=IN-1:GOTO
60200
60270 GOTO 60210
60280 PRINT TAB(JT+JW*(IN-1));MID$(JC$,Z
S,1);MID$(PR$,JT+JW*(IN-1)+1,JW);"T"
60290 RETURN
60500 GET Z$:IF Z$<>"Q"THEN RETURN
60600 GET Z$:IF Z$<>""THEN 60600
60605 POKE VIC+24,21:POKE VIC+21,0:PRINT
 CHR$(9);
60610 GOSUB 61000:POKE VIC+33,6:POKE VIC
+32,14:POKE SID+24,0:PRINT"":END
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(
13):SID=54272:JS=56320:CM=55296
61010 JC$="":QL=214:QI=255:RETURN
62000 L0=LEN(PG$)+2:L1=LEN(AU$)+2:L2=LEN
(A2$)+2:IF L1<L2 THEN L1=L2
62010 IF L0<L1+2 THEN L0=L1+2
62020 B0$=LEFT$("
                      ",L0+2)
62030 DEF FN T(N)=(40-N)/2:B1$=""+MID$(
B0$,2,L1+1):T1=FN T(L1):T0=FN T(L0)
62035 GOSUB 61000:POKE VIC+32,0:POKE VIC
+33,0:POKE SID+24,0
62040 PRINT"";CHR$(8):FOR I=1 TO 4:P
RINT TAB(T0);B0$:NEXT I
62050 PRINT"TT";TAB(FN T(LEN(PG$)));PG
$:PRINT
62060 FOR I=1 TO 3:PRINT TAB(T1);B1$:NEX
T I
62070 PRINT"T";TAB(FN T(LEN(AU$)));AU$

62080 IF A2$<>""THEN PRINT"";TAB(FN T(L
EN(A2$)));A2$:PRINT TAB(T1);B1$
62090 BG$="PULSE "+BG$+" P/ENTRAR":T0=FN
 T(LEN(BG$)+2)
62100 LN$=LEFT$("————————————————————
————————————————————",LEN(BG$))
62110 PRINT TAB(T0);"";LN$;""
62120 PRINT TAB(T0);"";LN$;""
62140 PRINT TAB(3);"COPYRIGHT (C) 1
984 THE CODE WORKS"
62150 L1=LEN(BG$):I=1:PRINT"TTTTTTTT"
62160 PRINT SPC(T0+1);MID$("",I,1);LEF
T$(BG$,L1);"T"
62170 L1=L1+1:IF L1>LEN(BG$)THEN L1=1:I=
3-I
62180 GET T$:IF T$<>""THEN 62200
62180 GET T$:IF T$<>""THEN 62200
62190 IF PEEK(JS)AND 16 THEN 62160
62200 CLR:GOSUB 61000:PRINT"":GOTO 100
```

CAMBIO DE IDENTIFICACION DE DISCO

COMP.: DREAN COMMODORE 64
CLAS.: UTI



Con este sencillo programa podremos cambiar la denominación de los discos, con sólo seguir las instrucciones que aparezcan en pantalla.

```
0 PRINT""
1 PRINTSPC(5)"ESTE PROGRAMA PERMITE C
AMBIAR"
2 PRINTSPC(10)"EL I.D.# DEL DISCO"
3 PRINTSPC(12)"PRESENTADO POR"
4 PRINTSPC(2)"BETTER  SOFTWARE & F.PINE
IRO Y LEONE"
5 PRINTSPC(12)"PULSE UNA TECLA"

6 GETA$:IFA$=""THEN6
10 OPEN9,0,0:OPEN15,8,15
15 POKE53280,14
16 POKE53281,1
30 MD$="":FORI=1TO32:MD$=MD$+"":NEXT
40 FORI=1TO39:BL$=BL$+" ":NEXT
50 P0$="CAMBIO DEL I.D.
#."
60 RE$="PULSE RETURN"
70 DATA2,FUERA DE RANGO,MUY EXTENSO.
80 READA:DIMEM$(A):FORI=1TOA:READEM$(I):
NEXT
99 GOTO1000
100 INPUT#15,ER:IFER=0THENRETURN
110 INPUT#15,ER,EM$,ET$,ES$
120 PRINTMD$"          ERROR EN DISCO!
#"ER" "EM$" "ET$", "ES$
130 END
200 INPUT#9,Q$:PRINT:Q1$=LEFT$(Q$,1):RET
URN
300 CLOSE2:CLOSE15:POKE59468,PK:PRINT""
:PRINT"FIN":END
400 PRINTLEFT$(MD$,MD)BL$:PRINTBL$
410 PRINTLEFT$(MD$,MD);:RETURN
500 MD=21:GOSUB400
510 PRINT"       ENTRADA INACEPTABLE--
"EM$(EM)
520 PRINTRE$:GOSUB200:GOSUB400
530 RETURN
1000 F=0:PRINTP0$
1020 DV=0
1040 PRINTP0$:PRINTSPC(4)"COLOQUE EL DI
SCO EN DISQUETERA"DV""
1050 PRINTRE$:GOSUB200
1060 PRINT#15,"I"+STR$(DV):GOSUB100
1060 PRINT#15,"I"+STR$(DV):GOSUB100
1070 OPEN2,8,2,"#":PRINT#15,"U1:2";DV;",
18,0":GOSUB100
1080 PRINT#15,"B-P:2,162":GOSUB100:DN$="
"
1090 FORI=1TO16:GET#2,A$:DN$=DN$+A$:NEXT

1100 MD=10:GOSUB400:IFFTHENRETURN
1110 PRINTSPC(7)"EL ID # DEL DISCO ES
";DN$
1130 MD=13
1170 MD=16
1175 PRINTSPC(9)"INGRESE";
1180 PRINT" NUEVO ID #":PRINTSPC(9)"LI
MITE 2 CARACTERES"CARACTERES
1190 GOSUB200:IFLEN(Q$)>2THENEM=2:GOSUB5
00:MD=18:GOSUB400:GOTO1190
1200 NDN$=LEFT$(Q$+BL$,2)
1210 MD=21:GOSUB400
1220 PRINTSPC(12)"SHALL I SEND "NDN$
1230 PRINTSPC(3)"AL DISCO EN DISQUETERA
"DV"? (S/N) ";
1230 PRINTSPC(3)"AL DISCO EN DISQUETERA
"DV"? (S/N) ";
1231 GETE$:IFE$=""THEN1231
1232 IFE$="N"THEN300
1240 GOTO1270
1250 IFQ1$<>"N"THENGOSUB400:MD=18:GOTO11
70
1260 GOTO1210
1270 MD=21:PRINTSPC(12)"SEA PACIENTE."
1280 PRINT#15,"B-P:2,162":GOSUB100
1290 PRINT#2,NDN$;:GOSUB100
1300 PRINT#15,"U2:2";DV;",18,0":GOSUB100
LIST 300-350
1310 PRINT#15,"I"+STR$(DV):GOSUB100:CLOS
E2
1320 F=1:PRINTP0$:GOSUB1060:F=0
1330 PRINTSPC(7)"EL NUEVO ID DEL DISCO E
S ";DN$
1350 CLOSE2:MD=21:GOSUB400:PRINTSPC(10)"
DESEA USTED PROCESAR"
1360 PRINTSPC(8)"ALGUN OTRO DISCO? (S/N)
 "
1350 CLOSE2:MD=21:GOSUB400:PRINTSPC(10)"
DESEA USTED PROCESAR"
1360 PRINTSPC(8)"ALGUN OTRO DISCO? (S/N)
 "
1361 GETU$:IFU$=""THEN1361
1370 IFU$="S"THEN1040
1380 IFU$="N"THEN300
1390 GOTO1361
```

GUIA PRACTICA



- SISTEMAS A MEDIDA
- ACCESORIOS
- BIBLIOGRAFIA
- JUEGOS PARA CASSETTES Y DISKETTES
- CURSOS PARA NIÑOS Y ADULTOS
- SERVICIO TECNICO TODAS LAS MARCAS

ZX SPECTRUM - TK - COMMODORE 64 Y 128 - TI 99 - 4 A

BS. AS. RUTA 202 Nº 1657
1611 - DON TORCUATO - TE. 748-2784

CORDOBA: 9 DE JULIO 70
LOCAL 2 - V.C. PAZ
TE. 22300

SITEC COMPUTACION

SERVICIO TECNICO GARANTIZADO
COMPUTADORAS - JOYSTICKS - TEXTOS
JUEGOS PARA CASSETTE Y DISKETTE
(DAM BUSTER, KARATE CHAMPS, RAMBO II, etc.)

VALENTIN GOMEZ 3521 - CAPITAL - 87-3512

ENVIOS AL INTERIOR

Para su Czerweny ahora si "Joystick"

Conecte su joystick directamente a su computadora CZ-1000, 1500, Spectrum y disfrute ya de su juego preferido

Adaptación + 1 joystick CZ-800 ₳ 19,.5
CZ-2000 SPECTRUM + 1 joystick + adaptación ₳ 259
CZ-1500 + 1 joystick + adaptación ₳ 148

INTELEC S.R.L.

Precio especial a distribuidores, adaptación autorizada por Czerweny "conserve su Garantía"
Llámenos o consulte a su distribuidor.

Paraná 426 2do. Cuerpo Of. 1 Cap. **40-7000**

— SERVICE INTEGRAL —

MICRODIGITAL
SINCLAIR - COMMODORE
REFORMAS A PAL-N C64/128/TK

LOGICAL LINE

URUGUAY 385 OF. 404 T.E.: 45-2688/5020
46-7915 INT. 404

NOVEDAD

RTTY COMMODORE 64 - 128 - TS 2068

INTERFACE - MODEM TRANSMISION y RECEPCION DE RTTY, BAUDOT, ASCII, CW. 45 A 300 BAUDIOS, SHIFT VARIABLE, BUFFERS. MODULO COMPACTO, ALIMENTACION DESDE LA COMPUTADORA, ETC. OFERTA ₳ 79

GALICIA 1279 1º "B" Tel.: 611-0505 **ENVIOS AL INTERIOR**

HALLEY COMPUTACION

CARTRIDGE EMULADOR SPECTRUM ₳ 35
INTERFACES PARA JOYSTICKS TIPO KEMPSTON PARA 2068 ₳ 25

descuentos especiales por mayor

Pueyrredón 232 - Tel.: 798-8480
MARTINEZ - Bs. As.

TODO PARA SU COMMODORE 64 Y 128 Y PC IBM



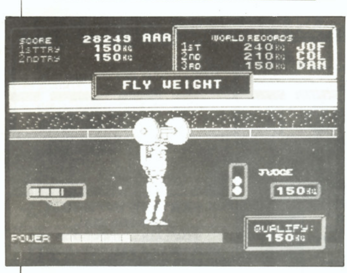
programas: *utilitarios*
juegos
manuales castellano

FORMULARIOS CONTINUOS - CINTAS IMPRESORAS DISKETTES 8" - 5 1/4" - 3,5"

SUMINISTROS OBELISCO CORRIENTES 1125 3º A 35-9614

atención especial a revendedores

HYPERSPORTS



CLASIFICACION: JUEGO DEPORTIVO
COMPUTADORA: SPECTRUM
CONFIGURACION: 48 K
FACTOR K 64: 10
HECHIZO: 9
DOCUMENTACION: N/D

Luego del rotundo éxito que rodeó al Decthalon, no podía esperarse mucho antes de que alguien se decidiera a sacar una segunda versión. Esta vez le tocó el turno a la empresa inglesa Imagine. Y no nos quepa ninguna duda que aquél refrán de "las segundas partes nunca fueron buenas" no tiene ninguna aplicación en este caso.

Nos encontramos con los seis eventos deportivos más emocionantes de una olimpíada, reproducidos con todo cuidado hasta el menor de sus detalles. Estos seis eventos son:

Natación: tenemos que lograr la clasificación nadando lo más rápidamente posible, haciendo ésto mediante las teclas adecuadas y apretando de tanto en tanto la tecla de aire. Los gráficos no son muy impresionantes, pero cumplen su cometido a la perfección.

Tiro al plato: En esta prueba, debemos derribar unos pequeños platos que aparecen en nuestro campo visual tanto desde la derecha como desde la izquierda. Nosotros debemos apuntar moviendo una mira en la pantalla y disparar cuando lo creamos oportuno. Los efectos gráficos son sencillos pero bien logrados y muy detallados.

Salto de potro: Aquí debemos correr a toda velocidad hacia el trampolín de salto y luego de darnos impulso y rebotar en el banco, hacer una pirueta en el aire y caer lo mejor que se pueda. Esta prueba tiene muy buenos gráficos.

Tiro con arco: Apuntaremos con cuidado y tomaremos en cuenta los efectos del viento para lograr un buen disparo. La prueba se ve simultáneamente desde dos frentes, ésto es, desde un costado y de frente.

Triple salto: Esta prueba consiste en un salto en largo, con la diferencia de que luego de picar una vez, podemos impulsarnos dos veces más. De esta forma se consigue un efecto más espectacular y es más fácil saltar.

Levantamiento de pesas: Esta prueba es la más difícil de todas. Debemos lograr aumentar lo suficiente la fuerza de nuestro competidor, pues de lo contrario no podremos levantar las pesas y perderemos una oportunidad.

Una vez que se completan exitosamente las pruebas debemos comenzar de nuevo, pero esta vez con marcas clasificatorias más altas.

Con Hypersports estamos frente a un juego muy entretenido, bien hecho y que sin duda alguna hará "sudar" a los teclados de las computadoras.

SABRE WULF



CLASIFICACION: JUEGO DE ACCION
COMPUTADORA: SPECTRUM/TK 90
CONFIGURACION: 48 K
FACTOR K 64: 7
HECHIZO: 8
DOCUMENTACION: 6

Luego de entretener y maravillar a sus fanáticos con las polvorientas cavernas de Atic Atac, Ultimate nos transporta ahora a una extraña jungla poblada de misterios y peligros que deberemos sortear.
El concepto de este juego es similar al de su predecesor (Atic Atac). Uno debe buscar las partes de un amuleto que nos permitirá salir de esta selva-laberinto. En el camino, deberemos enfrentarnos con distintas criaturas que nos atacarán, algunas de ellas débiles y otras casi invencibles. También nos encontraremos con muchos tesoros en nuestro camino.
La acción se desarrolla en una selva que consiste en un laberinto enorme, lleno de corredores y callejones sin salida. Los monstruos contra los que debemos defendernos se asemejan a hipopótamos, contra los cuales más vale esquivarlos que atacarlos. A propósito nuestra única arma es una espada y hasta haber adquirido algo de experiencia en su uso nos veremos morir una y otra vez, sin más remedio que recomenzar el juego una vez que se nos acaben todas las vidas.
Una de las características más divertidas consiste en las orquídeas profusamente coloreadas que encontraremos en nuestro camino. Sucede que si nos las comemos nos pasan cosas muy extrañas. Por ejemplo, podemos tener extraños y nuevos poderes.
Si bien los gráficos no están totalmente libres de error (todavía existen problemas cuando una figura coloreada pasa sobre otra) están de acuerdo a las normas de calidad de Ultimate, y esto ya son palabras mayores.
Sabre Wulf se trata de un juego rápido, que combina las características de un juego de acción y de aventuras, y que nos dará muchas horas de diversión sin aburrirnos.



¿APRENDER QUE?

Consultamos a quienes tienen a su cargo institutos de enseñanza de computación.

"No tienen que ser especialistas" Roberto Sade (ISMM)

Sin duda, la injerencia de la computación en la vida cotidiana es muy grande, pero hay que comprender que salvo aquellos que quieran ser profesionales en el tema, la mayoría lo que tiene que saber es la utilidad, las posibilidades y la utilización de la computación. No tienen que ser especialistas, no tienen que saber programar. Creo que a los chicos se les enseña mal. Se los mete en lenguajes haciéndoles perder un tiempo precioso en aprendizajes de técnicas que son viejas tan pronto como aprendidas. Al chico hay que familiarizarlo con la máquina, enseñarle a instrumentarla, que aprenda a usarla en su trabajo, etc. Hay una gran deshonestidad en hacerles creer a los padres que si los chicos no son técnicos en computación van a estar atrasados el día de mañana. En el fondo yo creo que hay un gran negocio en todo esto.

Manuel Schernitzki (DIPSA) *"Para desarrollar software"*

En cuanto al tema de educación de informática se ha producido una atomización total en un esfuerzo privado, disgregado, sin coherencia, sin idoneidad, tratando de captar el interés por la gran difusión que se le ha dado a través de los medios de difusión y la gran demanda que hay. Tanto las carreras oficiales de la Universidad como las terciarias están carentes de la necesidad real del mercado argentino. Se está tratando de educar pero en las tinieblas. En estos momentos hay cinco mil profesionales que dan vueltas sin poder ubicarse y que finalmente optan por irse del país

Por otra parte el ritmo vertiginoso del cambio de máquinas requiere una especialización rápida y mucha mano de obra, y las empresas se ven obligadas a requerir gente con experiencia porque no tienen tiempo de capacitar.

Lo que Argentina puede aportar al mundo son ideas, software, desarrollos. No podemos competir a nivel de chips o plaquetas con la tecnología de Japón, Singapur o Corea, por un problema de economía de escalas. Pero sí en la investigación o desarrollo del software de base. Tomemos como ejemplo que en Estados Unidos, los centros educativos de las universidades son los que desarrollan los paquetes de programas más famosos.

Daniel Piorun (CEI): *"Ayudar a razonar"*

En el campo de la educación la computadora debe ayudar a que la gente razone más, sea más creativa, expanda su mente y que no se encierre en un sistema educativo antiguo donde todo es cuadrado, recto y no deja mucho campo para discernir o pensar. Tal es el caso de nuestras universidades, que tienen un sistema de enseñanza bastante rígido.

En los chicos no hay miedos. A la primera hora de la primera clase para chicos uno ve cómo ya están metiendo y sacando. En cambio, en la misma situación, los adultos son todas momias. Es un poco el miedo a probar, a cambiar, a lo nuevo. Nosotros trabajamos en grupos de dos o tres personas por máquina y no porque nos falten computadoras sino porque queremos que se aprenda a trabajar en grupo.

Me parece bien que en las escuelas se enseñe computación, pero pienso que debe existir una infraestructura adecuada. Es decir, que no haya sólo máquinas sino también docentes capacitados en forma tal que puedan infundirle al chico la capacidad de pensar, razonar, discernir, crear y

José Alvarez (MUPIN): *"Que la computación sea una herramienta"*

El mercado puede ampliarse siempre y cuando se le de el apoyo necesario a nivel oficial y se permita el ingreso de nueva tecnología para que sea más accesible para el estudiantado. Lo fundamental es que se pueda dictar computación en la escuela primaria para que el niño vaya entrando al nuevo mundo. Que se refuerce en la escuela secundaria para que en los claustros universitarios la computación sea una herramienta.

Roberto Campos (Microcómputo): *"Materia obligatoria"*

Computación se debe enseñar como materia obligatoria desde tercer o cuarto grado. Logo en la primaria y Basic en la secundaria. Eso tiene que ser prioridad en las necesidades educacionales del país. Por suerte está bajando el precio del hard, pero igual estamos lejos de los valores internacionales.

Cuando iniciamos los cursos pensamos que los chicos tenían cierta receptividad. Pero la realidad superó ampliamente lo esperado. Tuvimos que profundizar mucho más la enseñanza porque por ser muy aptos y dúctiles los chicos exigían más. Además, hay una diferencia importante de resaltar. Cuando se equivoca en la escuela, el chico recibe una mala nota, cuando se equivoca en computación, la máquina le señala el error. Modificando el error va aprendiendo.

En la escuela lamentablemente seguimos con modelos superados de castigar el error. Hay que liberar la creatividad de los chicos. La enseñanza de adultos es más dificultosa. No existe tanta fluidez. Primero porque el adulto tiene una formación en la que no intervino la computadora. En general son profesionales, con un elevado coeficiente intelectual, pero a los que les cuesta mucho enfrentar a la computadora.

Marcelo Capelli: *"Idoneidad para enseñar"*

Habría que ver qué grado de idoneidad tienen los que enseñan



Se conocen casos de personas que han comprado una computadora, estudiaron el manual y se pusieron a dar clases de computación. En los casos de Logo por ejemplo, al chico se le dan instrucciones primitivas para que empiece a hacer sus preguntas con demasiado tiempo libre.
Los chicos son como esponjas, absorben todo el conocimiento.
Se sienten desafiados a hacer una cosa y cuando lo logran se sienten premiados por el resultado que pueden palpar inmediatamente.
La computadora le da la posibilidad de autosuperarse. Lo que veo es que los chicos se ponen a trabajar con una computadora y dejan los juegos colectivos donde hay cooperación.
Es un desafío individual, que desde el punto de vista social crea una personalidad especial, en la que a partir de un razonamiento lógico lo que está bien, está bien y se premia, y lo que está mal, no. De esta manera se crea un SI y un NO sin darle al chico la posibilidad de que en el medio tenga toda la gama de grises. El razonamiento lógico es muy útil para la ciencia, pero sería bueno que el chico tenga toda la "gama de grises" que la computadora no le da. Por eso, la computación debe complementarse con la parte afectiva y de comprensión entre el adulto y el chico. Y en ese sentido es importante que los padres se introduzcan en el mundo de la computación

La computación es una herramienta para aplicar en cualquier carrera. Triunfará el que pueda y sepa aplicar esa herramienta en su especialidad. El que domine la herramienta será especialista en esa herramienta y nada más. La gente se encandila pensando que el analista tiene el porvenir en sus manos, pero tiene sólo el dominio de una herramienta.

Instituto Superior Mariano Moreno

El año pasado de los doce mil alumnos del Instituto, diez mil fueron de computación. Están equipados con IBM, Texas Instruments, Xerox, etc. La novedad es la computación para niños porque este es un instituto individualizado como "de mayores".
Para 1986 tendran Cursos de Capacitación para Docentes, con instructores de primer nivel

Microcómputo

En una primera instancia usaron equipos TI-99/4A. Posteriormente, la venta y demanda de otros equipos los llevó a incorporar otras máquinas.
Dividieron la enseñanza de programación y lenguaje Basic en dos momentos. El primero es para principiantes, es un curso donde se brindan conocimientos básicos, de diagramación y programación. El segundo curso requiere la aprobación del primero y es más profundo. Se hace manejo de archivo, periféricos, gráficos y demás. Cada curso dura tres meses, salvo en verano, cuando se dictan cursos intensivos. También incursionaron en la enseñanza de Assembler (fundamentalmente para Texas, Sinclair y C64).

EDUCACION

DIPSA (Difusión Informática Profesional)

DIPSA da cursos para profesionales. Este año incorporó un sector, DIPSA Junior, para chicos. Se dan a través de empresas de primera línea, ya sea para el personal o a sus hijos. Por ahora no tiene una carrera oficial pero está dentro de los planes incorporarla como así también un Master de Estados Unidos. Los lenguajes están divididos en módulos de acuerdo a la capacidad y necesidad del alumno (Basic, Cobol, Algol, Pascal y Logo para docentes), además de Análisis de Sistema como un curso introductorio y otro prolongado. Se enseña con televideo, que es compatible con IBM.

Capelli

Se dan cursos de Logo, Basic, que están separados en dos niveles: el Basic Fundamental y el Extendido. Por Fundamental se entiende el concepto integrado de hardware y software de la computadora en conjunto. No sólamente se inicia con instrucciones del Basic sino que también se empieza con "Qué es una computadora", desde la necesidad de su creación, los primeros rudimentos, hasta un diagrama en bloques constitutivos de las distintas partes esenciales de una computadora vista por dentro.

Las computadoras que se utilizan son de eras y tecnologías distintas, como la Texas y la Commodore y opcionalmente.

CEI (Centro de Educación Informática)

Hay distintos tipos de cursos en la sede central y sucursales. Además dan cursos en lugares específicos como por ejemplo bancos que solicitan capacitar un área, o una asociación de psicólogos, etcétera.
Los cursos normales son los de Logo, que es un lenguaje para chicos de escuela primaria y cuya característica principal es que les permite a los niños tener un amplio grado de creatividad.
Luego tienen cursos de Basic, que es para adolescentes y adultos, en el que la complejidad es mayor y las posibilidades importantes. Están orientados por edades y en área comercial existe una gran demanda. Otros están dedicados al lenguaje Assembler, que ya es más avanzado.
Aparte hay cursos de Programas Utilitarios

Instituto MUPIN

Ofrece cursos oficiales de Analista Programador con una duración de dos años y luego Analista de Sistema con una duración de un año más. Son títulos oficiales y los egresados están habilitados para cumplir funciones en empresas públicas, privadas y en cualquier organismo que aplique sistemas de procesamiento de datos.
Al mismo tiempo se dan cursos breves para capacitación de inscriptos que no conozcan nada sobre el tema (de Basic, Cobol, Operador Sistema 34 y de Introducción al análisis de Sistema, Pascal y otros).
Poseen una IBM Sistema 34.



GUIA PRACTICA

Compatibilidad de ROM

A fuerza de dejar de ser original quisiera felicitarlos por su revista.
Hace más de un año y medio que tengo una TS 2068 con impresora térmica y trato de adquirir todo el material que puedo, incluídas algunas revistas españolas que andan dando vueltas y que no son específicas para mi máquina, sino para la SPECTRUM.
Al mismo tiempo, tengo el mapa con las direcciones de la rom de mi máquina y quisiera preguntarles en dónde puedo conseguir esta información de la SPECTRUM, pues, varios programas dejan de andar porque llamo a una rutina que existe pero en una dirección distinta (tengo entendido que gran parte de las ROM's de ambas máquinas son esencialmente iguales, pero distribuídas en distinta forma).
Recurro a Ustedes, pues me he cansado de preguntar a personas que gozan retaceando información.
Sigan como hasta ahora, y desde ya, gracias.
Mario José Droszler
Guevara 71
(1427) Capital

K64:
Lo que decís en tu carta, Mario, es correcto en todo sentido. Tanto en cuanto a las ROM's de la SPECTRUM y la 2068 como en lo que se refiere a ese tipo de gente que experimenta un cierto placer sádico retaceando información. Es una pena que este tipo de gente abunde en nuestro medio, pero este es un problema que no nos toca analizar en estas líneas.

En cuanto a tu pedido específico, hemos confeccionado una lista que menciona las rutinas de la 2068, su equivalente en la SPECTRUM y las direcciones en las rom's de las mismas. Si bien pueden ser ligeramente distintas en su forma, llevan a cabo los mismos cometidos.

No nos cabe duda que esto será de gran ayuda para todos aquellos que quieran adaptar programas entre ambas máquinas.

Al respecto de la información de la rom de la SPECTRUM sabemos que existe un libro publicado por Melbourne House y escrito por Ian Logan llamado "The Complete Spectrum Rom Dissasembly" en donde se explica paso a paso y mediante la separación en rutinas específicas los 16 K de la ROM de la SPECTRUM. Esperamos haberte sido de ayuda y no te desanimes por los "retaceadores". Seguí adelante!

EL UNICO MEDIO
REVISTAS
CON ALCANCE NACIONAL

La Asociación Argentina de Editores de Revistas ha propuesto para su publicación —en espacios preferenciales— de este emblema distintivo que reafirma y reconoce a la revista, como el único medio de comunicación con alcance nacional. Alcance que conlleva el concepto de integración, uno de los objetivos propuestos de mayor importancia en lo que a medios de comunicación se refiere.

Programador del año

Tengo interés en participar en el concurso del programador del año. Tengo una TANDY 1000 128K (compatible con IBM PC). Por lo tanto quisiera me aclarasen si existen limitaciones en cuanto al tipo de procesador y capacidad de memoria de la máquina utilizada ya que no figura en las bases. En caso de no poder participar con un programa para dicha máquina quisiera saber si puedo hacerlo con uno para la RADIO SHACK TRS-80 64 K COLOR COMPUTER 2 con BASIC EXTENDIDO.
Gabriel A. Moreno
San José - Mendoza

K64:
En cuanto al tipo de máquina con la que se puede intervenir en el concurso, no hicimos referencia a nombres en par-

ticular pero damos por entendido que se trata de Home Computers y no de Personal Computers. Por Home Computers entendemos máquinas que no superen los 64 k de RAM del tipo de las que nos ocupamos con más profundidad en nuestra publicación.
Espero que sepas entender esta condición, dado que sería poco honesto hacer competir un programa para una CZ-1500 con 16K con uno hecho en un IBM PC.
Por lo antes mencionado no existe inconveniente alguno en que nos mandes tu programa hecho en la COCO.

Amigos de K64:

Desde ya mis felicitaciones por el excelente material que brindan mes a mes.
Bárbaras las notas sobre robótica e informática, todas de actualidad.
Muy buena idea de publicar varios métodos para "entrar" en programas cerrados.
Podrían sacar algún método para ingresar en programas, pero para máquinas como la SPECTRUM.
Sin más, los saludo y les deseo la mejor de las suertes.

Hernán Polo
Humberto 1 - Sta. Fe

K64:
Te agradecemos las felicitaciones y tendremos en cuenta tu sugerencia.

DEBUGGING



Cometa Halley

Vuelto a distraer su atención en poco tiempo, espero ser disculpado, esta vez es para señalar una serie de errores cometidos en el programa COMETA HALLEY.
Los errores son los siguientes:
1) La línea 30 debe ser LET X(7)=100 (valor aproximado) y no como se publicó; 2) La línea 92 es LET M(22)=4.3 y la línea 93 es LET F$(23)="15/4", faltaba en el programa publicado la línea 93; 3) Falta la línea 133, que es LET M(32)=5; 4) En la línea 119 no es=111 sino =11, por eso daba un error B/1100; 5) Las líneas 370 y 740 tienen un 200 como divisor cuando lo correcto es un 260, éste era el otro error B(745); 6) La línea 30 del programa publicado es en realidad la línea 330; y 7) La línea 1100 es UNPLOT y la 1110 es PLOT y no como se publicó en "K64".
Ajusté las coordenadas de los ejes para ver el Halley desde la ciudad de Buenos Aires (latitud sur 34 y longitud oeste 58).
Ah, también eliminé los REM de las líneas 141 y 145, así el cartel de presentación se arma más rápido.
Posdata: El procesador de textos funciona a las mil maravillas. Muchas gracias por publicarlo.
Hasta la próxima...

Mauricio G. Uldane
Capital Federal

Bolsa de Usados

• Deseo intercambiar programas para la TK 83 con 16 Kbytes.
Adrián E. del Campo 2364 Quilmes Oeste (1879) - Pcia. Bs. As.

• Vendo computadora TK 83 con expansor de 16 K, Joystick, manual en castellano y muchos programas de 2 y 16 k. Excelente estado. Precio a convenir. TE.: 760-0370

• Vendo CZ 1000 en buen estado con 5 cassette de juegos y un Libro de siete juegos. ₳ 100. TE: (0954) 94145. Marcelo Bajo-Estanislao Ceballos 227 - Conquimay - La Pampa

• Vendo interfase 1, microdrive y 4 cartuchos sin uso. También impresora Alphacom 32.
TE: 825-2302

• Vendo TK 85 en perfecto estado con grabador, joystick, 18 juegos dos libros y manual. Todo ₳ 190 TE: 765-0295. Av. Sucre 2158. San Isidro.

• Vendo computadora TS 1000 impecable con más de 40 programas (juegos y utilitarios). TE: 208-7415

• Vendo Cartridge emulador Spectrum para TS 2068. TE: 825-2302



PREMIOS DEL MES

COMPUTADORAS - CASSETTES - BECAS

SUSCRIPTORES

GANADOR DEL MES, SORTEO CZ 1000

JUAN M . TABORDA

Wilde - Pcia. Bs. As.

FELICITACIONES

DEBERAS RETIRARLA EN NUESTRAS OFICINAS CON LA PRESENTACION DEL D. DE IDENTIDAD

SORTEO ENCUESTA: GANADORES DEL MES

PREMIO: 10 BECAS

DIEGO BENITEZ
DIEGO ESTEVE
ALEJANDRO BUSTI
MARIANO A. PANDO
GUIDO E. BENHERGUI
CARLOS G. SPOSARO
GASTON MIHITAR
MANUEL DE OLASO
ALBERTO QUINTA
JUAN J. HERNANDEZ

PREMIO: 40 CASSETTES

PABLO PRIETO	Rio Negro
YACO A. MAZAL	Misiones
DANIEL DZIADEK	Neuquén
CESAR MONTANARO	Santa Fe
DANIEL QUADRELLI	V. Tuerto S.F.
JOSE A. CALVO	Entre Rios
IVAN E. GAITAN	Misiones
HORACIO A. STURZENBAUM	Santa Cruz
FRANCISCO MONTERO	S. Miguel de Tucumán
DIEGO ATHOS CORMACE	Rio Negro
HORARIO RODRIGUEZ	La Pampa
SERGIO A. CERVIÑO	Chaco
ROQUE SARDA	Mendoza
MATIAS PALMEIDA	Chubut
ESTEBAN DURAND	San Juan
CARLOS BOTTCHER	Catamarca
GONZALO F. RUBIES	Mendoza
JORGE D. SOJA	La Pampa
OSCAR A. MOYANO	San Luis
DANIEL R. RAMIREZ	Corrientes
SILVINA FONTANA	Córdoba
M. SONIA ZUM	Córdoba
CARLOS O. OTALORA	La Pampa
FANNY B. AZAR	Formosa
JUAN C. FOURCADE	S.C. de Bariloche
RODRIGO BRIZUELA	La Rioja
LUIS O. DACUNDA	Misiones
MARIA E. LOPEZ	Tierra del Fuego
VICTOR M. PINOS	Formosa
CARLOS VALLEJOS	Corrientes
SERGIO E. DROLINA	S.S. de Jujuy
CESAR H. WILLIAM	Tierra del Fuego
EDUARDO CATENA	S.M. de Tucumán
OMAR MONTE	Córdoba
GUILLERMO R. CARO	Neuquén
ESTEBAN JAIMEZ	San Luis
HERNAN ELENA	Rio Negro
ARIEL ANTIK	Rosario
EDUARDO M. SARRA	Córdoba
ORLANDO F. CARRILLO	San Luis

Los premios podrán retirarse en la Administración de K64, Cerrito 1320 - 1º P. Capital, con documento de identidad en el horario de 10 a 12 y 15 a 17 hs. Quienes viven en el interior del país, pueden solicitar que se les remitan los premios por correo.


Load
REVISTA MENSUAL
PARA USUARIOS DE
MARZO 1986
Nº 1 # 2
sinclair
Programas
INEDITOS PARA
CZ 1000 1500 2000
TK 83 85 90X
APLICACIONES COMERCIALES Y EDUCATIVAS
Z-80: EL REY DE LOS PROCESADORES
GENESIS DE LA PROGRAMACION
Gráficos
LOS MONSTRUOS MAS FAMOSOS